bytecluster0001

bytecluster0001 ist ein virtueller Server, der Kommunikationsdienste für den Verein bereitstellt. Der Server wurde von der Firma Hetzner Online GmbH dankenswerter Weise zur Verfügung gestellt.

Administratoren

- mape2k
- mkzero
- suicider

Benutzer

- Bernd (Webseiten)

IPs /DNS

- bytecluster0001.bytespeicher.org
  - 88.198.111.196
  - 2a01:4f8:c17:1214::2

Installation

- Debian 8.2 minimal

User / Gruppen

- mkzero → sudo
- marcel → sudo
- stephan → sudo
- bernd → sudo für www-data
- bytebot
- twitterstatus
- twitterstatus-ms
- spacestatus
- redmine
- ffapi
- synapse

Pakete

- zsh
- git
- screen
- mosh (SSH via UDP)
- python
- mc
- debian-goodies

Netzwerk

Skript für IPv6-Adressen (benötigt für Matrix-IRC-Bridge)

/usr/local/bin/manage_ipv6_addresses.sh

```
#!/bin/bash

ACTION=$1
```

```
BASEADDR=$2
NETMASK=$3
COUNT=$4
INTERFACE=$5

for i in $(seq 1 $COUNT); do
  ip -6 address $ACTION $(printf "%s:%04x/%s" $BASEADDR $i $NETMASK) dev $INTERFACE
done
```

- ***chmod +x /usr/local/bin/manage_ipv6_addresses.sh***

Konfiguration

/etc/network/interfaces

```
# Loopback device:
auto lo
iface lo inet loopback

# device: eth0
auto  eth0
iface eth0 inet dhcp

iface eth0 inet6 static
  address 2a01:4f8:c17:1214::2
  netmask 64
  gateway fe80::1
  # 128 IPv6-Addressen mit Prefix "2a01:4f8:c17:1214::1:/64" anlegen
  post-up /usr/local/bin/manage_ipv6_addresses.sh add 2a01:4f8:c17:1214::1 64 128 eth0
  pre-down /usr/local/bin/manage_ipv6_addresses.sh delete 2a01:4f8:c17:1214::1 64 128 eth0
```

Konfiguration SSH

- HostKey DSA entfernt
- PermitRootLogin no
- PasswordAuthentication no

/etc/motd

```
 ______   _______ _____ ____  ____  _____ ___ ____ _   _ _____ ____
| __ ) \ / /_   _| ____/ ___||  _ \| ____|_ _/ ___| | | | ____|  _ \
|  _ \\ V /  | | |  _| \___ \| |_) |  _|  | | |   | |_| |  _| | |_) |
| |_) || |   | | | |___ ___) |  __/| |___ | | |___|  _  | |___|  _ <
|____/ |_|   |_| |_____|____/|_|   |_____|___\____|_| |_|_____|_| \_\

    > BYTECLUSTER0001
```

SUDO

- Administrative Benutzer sind Mitglied der Gruppe „sudo“

IPTABLES

- iptables-persistent

/etc/iptables/rules.v4

```
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [0:0]

# Already opened connections
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

# Garbage
-A INPUT -m state --state INVALID -j DROP

# Ping
-A INPUT -p icmp -m icmp --icmp-type 8 -j ACCEPT

# Localhorst
-A INPUT -s 127.0.0.0/8 -j ACCEPT

# Turnserver
-A INPUT -p udp -m udp --dport 3478 -j ACCEPT
-A INPUT -p udp -m udp --dport 5349 -j ACCEPT
-A INPUT -p udp -m udp --dport 49152:59999 -j ACCEPT

# SSH / mosh
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p udp -m udp --dport 60000:60008 -j ACCEPT

# Webserver
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT

# Mail
-A INPUT -p tcp --dport 25 -j ACCEPT
-A INPUT -p tcp --dport 110 -j ACCEPT
-A INPUT -p tcp --dport 143 -j ACCEPT
-A INPUT -p tcp --dport 465 -j ACCEPT
-A INPUT -p tcp --dport 587 -j ACCEPT
-A INPUT -p tcp --dport 993 -j ACCEPT
-A INPUT -p tcp --dport 995 -j ACCEPT
-A INPUT -p tcp --dport 2000 -j ACCEPT
-A INPUT -p tcp --dport 4190 -j ACCEPT

# Matrix
-A INPUT -p tcp -m tcp --dport 8008 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8448 -j ACCEPT
COMMIT
```

/etc/iptables/rules.v6

```
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [0:0]

# Localhorst
-A INPUT -i lo -j ACCEPT

# Piing
-A INPUT -p ipv6-icmp -j ACCEPT

# Already opened connections
```

```
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

# Garbage
-A INPUT -m state --state INVALID -j DROP

# Turnserver
-A INPUT -p udp -m udp --dport 3478 -j ACCEPT
-A INPUT -p udp -m udp --dport 5349 -j ACCEPT
-A INPUT -p udp -m udp --dport 49152:59999 -j ACCEPT

# SSH / mosh
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p udp -m udp --dport 60000:60008 -j ACCEPT

# Webserver
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT

# Mail
-A INPUT -p tcp --dport 25 -j ACCEPT
-A INPUT -p tcp --dport 110 -j ACCEPT
-A INPUT -p tcp --dport 143 -j ACCEPT
-A INPUT -p tcp --dport 465 -j ACCEPT
-A INPUT -p tcp --dport 587 -j ACCEPT
-A INPUT -p tcp --dport 993 -j ACCEPT
-A INPUT -p tcp --dport 995 -j ACCEPT
-A INPUT -p tcp --dport 2000 -j ACCEPT
-A INPUT -p tcp --dport 4190 -j ACCEPT

# Matrix
-A INPUT -p tcp -m tcp --dport 8008 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8448 -j ACCEPT
COMMIT
```

MySQL/MariaDB

- mariadb-server

/etc/mysql/my.cnf.patch

```
--- /etc/mysql/my.cnf.dist  2015-11-04 22:19:31.589007928 +0100
+++ /etc/mysql/my.cnf   2015-11-04 22:19:31.577007958 +0100
@@ -36,6 +36,9 @@
 skip-external-locking

 bind-address       = 127.0.0.1
+
+default_storage_engine = InnoDB
+
 #
 # * Fine Tuning
 #
@@ -68,6 +71,22 @@
 #long_query_time = 2
 #log_queries_not_using_indexes

+table_cache            = 500
+query_cache_limit  = 4M
+query_cache_size        = 128M
+
```

```
+# INNODB PERFORMANCE
+innodb_buffer_pool_size         = 256M
+innodb_log_buffer_size          = 8M
+innodb_log_file_size            = 128M
+
+innodb_log_files_in_group       = 2
+innodb_flush_log_at_trx_commit  = 2
+innodb_flush_method             = O_DIRECT
+innodb_file_per_table
+
+innodb_thread_concurrency       = 8
+
 [mysqldump]
 quick
 quote-names
```

NGINX

- nginx

/etc/nginx/conf.d/ssl.conf

```
ssl_protocols TLSv1.2;
ssl_prefer_server_ciphers on;

ssl_ciphers "EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH:!AES128";
ssl_ecdh_curve secp384r1; # Requires nginx >= 1.1.0

ssl_session_cache shared:SSL:10m;
ssl_session_timeout 10m;

ssl_session_tickets off; # Requires nginx >= 1.5.9
ssl_stapling on; # Requires nginx >= 1.3.7
ssl_stapling_verify on; # Requires nginx => 1.3.7

#add_header Strict-Transport-Security "max-age=63072000; includeSubdomains; preload";
add_header X-Frame-Options DENY;
add_header X-Content-Type-Options nosniff;

resolver 213.133.98.98 213.133.99.99 valid=300s;
resolver_timeout 5s;
```

/etc/nginx/patch

```
diff -Naur /etc/nginx.dist/nginx.conf /etc/nginx/nginx.conf
--- /etc/nginx.dist/nginx.conf  2014-12-01 12:12:00.000000000 +0100
+++ /etc/nginx/nginx.conf   2015-11-04 22:42:03.837950276 +0100
@@ -30,8 +30,8 @@
    # SSL Settings
    ##

-   ssl_protocols TLSv1 TLSv1.1 TLSv1.2; # Dropping SSLv3, ref: POODLE
-   ssl_prefer_server_ciphers on;
+   #ssl_protocols TLSv1 TLSv1.1 TLSv1.2; # Dropping SSLv3, ref: POODLE
+   #ssl_prefer_server_ciphers on;

    ##
    # Logging Settings
@@ -45,7 +45,7 @@
```

```
    ##

    gzip on;
-   gzip_disable "msie6";
+   #gzip_disable "msie6";

    # gzip_vary on;
    # gzip_proxied any;
```

Let's Encrypt (SSL-Zertifikate)

Installation

- ***useradd letsencrypt -m -G www-data***
- ***su - letsencrypt***
- ***git clone https://github.com/lukas2511/letsencrypt.sh.git***
- ***cd letsencrypt.sh***
- ***cp docs/examples/* ./***
- ***chmod ug+x hook.sh***
- ***mkdir /home/letsencrypt/letsencrypt.sh/.acme-challenges***

/etc/sudoers.d/letsencrypt

```
# Allow reload of NGINX
letsencrypt ALL=NOPASSWD: /bin/systemctl reload nginx.service
# Allow restart of Postfix/Dovecot
letsencrypt ALL=NOPASSWD: /bin/systemctl restart postfix.service
letsencrypt ALL=NOPASSWD: /bin/systemctl restart dovecot.service
```

Konfiguration Let's Encrypt-Client

/home/letsencrypt/letsencrypt.sh/config

```
CA="https://acme-v01.api.letsencrypt.org/directory"
...
CHALLENGETYPE="http-01"
...
KEYSIZE="4096"
...
HOOK=${SCRIPTDIR}/hook.sh
...
RENEW_DAYS="60"
...
PRIVATE_KEY_RENEW="yes"
...
KEY_ALGO=rsa
...
CONTACT_EMAIL=hostmaster@bytespeicher.org
```

/home/letsencrypt/letsencrypt.sh/hook.sh

```
function deploy_cert {
    # Reload NGINX
    sudo /bin/systemctl reload nginx.service

    # Copy erfurt.chat-Certificate/Key to synapse-directory
```

```
    if [ ${DOMAIN} = "erfurt.chat" ]; then
      cp -L ${KEYFILE} /home/synapse/ssl/
      cp -L ${CERTFILE} /home/synapse/ssl/
      cp -L ${FULLCHAINFILE} /home/synapse/ssl/
      chgrp synapse /home/synapse/ssl/*.pem
      chmod 640 /home/synapse/ssl/*.pem
    fi

    # Restart Postfix/Dovecot
    [ ${DOMAIN} = "mail.bytespeicher.org" ] && (sudo /bin/systemctl restart postfix.service;
sudo /bin/systemctl restart dovecot.service)
}
```

Konfiguration NGINX

/etc/nginx/snippets/letsencrypt.conf

```
# Use acme-challenge directory from letsencrypt.sh
location ^~ /.well-known/acme-challenge/ {
    default_type "text/plain";
    alias /home/letsencrypt/letsencrypt.sh/.acme-challenges/;
}

# Hide using ACME-Client
location = /.well-known/acme-challenge/ {
    return 404;
}
```

/etc/crontab

```
# Let's Encrypt
23 4 * * *     letsencrypt  /home/letsencrypt/letsencrypt.sh/letsencrypt.sh -c >
/home/letsencrypt/letsencrypt.log 2>&1
```

Verwendung des Let'sEncrypt Client für eine neue Domain

Pro Zertifikat können mehrere Domains/Subdomains integriert werden. Diese müssen in der domains.txt in einer Zeile stehen.

1. Let's Encrypt ACME-Challenge-Verifikation im VHost aktivieren

   /etc/nginx/sites-available/example.org

   ```
   server {
     ...
     include snippets/letsencrypt.conf;
     ...
   }
   ```

2. Domain eintragen und Zertifikat erzeugen

   /home/letsencrypt/letsencrypt.sh/domains.txt

   ```
   example.org www.example.org
   ```

   - ***su - letsencrypt***
   - ***cd letsencrypt.sh***
   - ***./letsencrypt.sh -c***

3. Verbindung als Nutzer beenden
   - ***exit***
4. DH-Parameter erstellen
   - ***mkdir /etc/ssl/example.org***
   - ***openssl dhparam -out /etc/ssl/example.org/dhparam.pem 4096***
5. SSL mit HSTS aktivieren und SSL-Zertifikate im NGINX einbinden

   /etc/nginx/sites-available/example.org

```
server {
  ...
  ssl on;

  add_header Strict-Transport-Security "max-age=31536000";
  add_header X-Frame-Options SAMEORIGIN;

  ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/example.org/fullchain.pem;
  ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/example.org/privkey.pem;
  ssl_dhparam /etc/ssl/example.org/dhparam.pem;

  ssl_trusted_certificate
/home/letsencrypt/letsencrypt.sh/certs/example.org/fullchain.pem;
  ...
}
```

6. NGINX neuladen
   - ***systemctl reload nginx.service***

PHP

- php5-fpm
- php5-curl
- php5-imap
- php5-gd
- php5-intl
- php5-mcrypt
- php5-json
- php5-mysqlnd
- php5-memcached
- php5-xmlrpc

/etc/php5/fpm/conf.d/50-local.ini

```
[Date]
date.timezone = "Europe/Berlin"

[PHP]
upload_max_filesize = 64M
post_max_size = 64M
```

Ruby

- ruby

Bytebot

Pakete:

- python-pip
- virtualenv

- python-dev (virtualenv build dep)
- libjpeg-dev (virtualenv build dep)
- zlib1g-dev (virtualenv build dep)
- libffi-dev (virtualenv build dep)
- libssl-dev (virtualenv build dep)

Installation:

/etc/systemd/system/bytebot.service

```
[Unit]
Description=Bytespeicher IRC bot
After=network-online.target
After=syslog.service
Requires=network-online.target
Requires=syslog.service

[Service]
User=bytebot
Group=bytebot
Restart=always
RestartSec=30
ExecStart=/home/bytebot/Bytebot/env/bin/python /home/bytebot/Bytebot/bytebot.py
MemoryLimit=256M

[Install]
WantedBy=multi-user.target
```

- ***sudo -u bytebot /bin/bash***
- ***cd /home/bytebot***
- ***git clone https://github.com/Bytespeicher/Bytebot***
- ***cd Bytebot***
- ***virtualenv env***
- ***. env/bin/activate***
- ***pip install -r contrib/requirements.txt***
- ***systemctl enable bytebot.service***
- ***systemctl start bytebot.service***

Twitterstatus / Twitterstatus Makerspace

Die Anleitung ist für „twitterstatus“. Die Einrichtung von „twitterstatus-ms“ erfolgt

Pakete:

- python-pip
- virtualenv

Installation:

- ***useradd -m twitterstatus***
- ***sudo -u twitterstatus /bin/bash***
- ***cd /home/twitterstatus***
- ***mkdir tmp***
- ***git clone https://github.com/Bytespeicher/twitterstatus***
- ***cd twitterstatus***
- ***cp config.py{.example,}***
- ***nano config.py***

~/twitterstatus/config.py

```
OAUTH_TOKEN     = '...'
OAUTH_SECRET    = '...'
CONSUMER_KEY    = '...'
CONSUMER_SECRET = '...'
ADMIN_NAME      = 'TWITTER_ACCOUNT_NAME_OF_ADMIN'
STATUS_FILE     = '/home/twitterstatus/tmp/twitter_old_status'
CURRENT_STATUS  = '/home/twitterstatus/tmp/status.json'
```

- ***virtualenv env***
- ***. env/bin/activate***
- ***pip install Twitter***
- ***exit***

/etc/systemd/system/twitterstatus.service

```
[Unit]
Description=Bytespeicher Twitter status bot
After=network-online.target
After=syslog.service
Requires=network-online.target
Requires=syslog.service

[Service]
User=twitterstatus
Group=twitterstatus
Restart=always
RestartSec=60
ExecStart=/home/twitterstatus/twitterstatus/env/bin/python
/home/twitterstatus/twitterstatus/bytebot.py
MemoryLimit=64M

[Install]
WantedBy=multi-user.target
```

- ***systemctl enable twitterstatus.service***
- ***systemctl start twitterstatus.service***
- ***crontab -u twitterstatus -e***

crontab -u twitterstatus -e

```
MAILTO=""
* * * * * /usr/bin/wget http://status.bytespeicher.org/status.json -O
/home/twitterstatus/tmp/status.json
```

Freifunk-API

Pakete

- python

Installation

- ***mkdir -p /var/www/api.erfurt.freifunk.net/public_html/***
- ***touch /var/www/api.erfurt.freifunk.net/public_html/freifunk-api.json***
- ***chown -R www-data:www-data /var/www/api.erfurt.freifunk.net/***
- ***chmod -R g+w /var/www/api.erfurt.freifunk.net/***
- ***useradd -m -G www-data ffapi***
- ***sudo -u ffapi /bin/bash***

- ***cd /home/ffapi***
- ***git clone https://github.com/FreifunkErfurt/ffapi***
- ***git clone https://github.com/FreifunkErfurt/scripts/ ffapi-update***
- ***cp ffapi-update/ffapi/config.py.example ffapi-update/ffapi/config.py***

Konfiguration

~/ffapi-update/ffapi/config.py

```
BASE_URL = 'http://map.erfurt.freifunk.net'
API_FILE_TEMPLATE = "/home/ffapi/ffapi/ff-erfurt.json"
API_FILE = "/var/www/api.erfurt.freifunk.net/public_html/freifunk-api.json"
```

Test

- ***ffapi-update/ffapi/ffapi-update.py***

  ffapi-update/ffapi/ffapi-update.py

  ```
  Update of /var/www/api.erfurt.freifunk.net/public_html/freifunk-api.json successful.
  We now have 146 Nodes
  ```

- ***logout***

Konfiguration Webserver

/etc/nginx/sites-available/api.erfurt.freifunk.net

```
server {
  listen      80;
  listen [::]:80;
  listen      443 ssl;
  listen [::]:443 ssl;

  server_name api.erfurt.freifunk.net;

  include snippets/letsencrypt.conf;
  if ($scheme != "https") {
    rewrite ^ https://$host$uri permanent;
  }

  ssl on;

  add_header Strict-Transport-Security "max-age=31536000";
  add_header X-Frame-Options SAMEORIGIN;

  ssl_certificate
/home/letsencrypt/letsencrypt.sh/certs/api.erfurt.freifunk.net/fullchain.pem;
  ssl_certificate_key
/home/letsencrypt/letsencrypt.sh/certs/api.erfurt.freifunk.net/privkey.pem;
  ssl_dhparam /etc/ssl/api.erfurt.freifunk.net/dhparam.pem;

  ssl_trusted_certificate
/home/letsencrypt/letsencrypt.sh/certs/api.erfurt.freifunk.net/fullchain.pem;

  gzip on;
  gzip_disable "msie6";
```

```
  gzip_vary on;
  gzip_proxied any;
  gzip_comp_level 6;
  gzip_buffers 16 8k;
  gzip_http_version 1.1;
  gzip_types text/plain text/css application/json application/x-javascript text/xml
application/xml application/xml+rss text/javascript;

  client_max_body_size 16m;

  location / {
    root   /var/www/api.erfurt.freifunk.net/public_html/;
    index  index.php index.html index.htm;
    autoindex on;
  }

  access_log /var/log/nginx/api.erfurt.freifunk.net-access.log;
  error_log /var/log/nginx/api.erfurt.freifunk.net-error.log;
}
```

- ***cd /etc/nginx/sites-enabled/***
- ***ln -s ../sites-available/api.erfurt.freifunk.net api.erfurt.freifunk.net***

Aktivierung Webserver

- alle SSL-Direktiven in der Konfiguration müssen kommentiert werden
- ***systemctl reload nginx***
- nun muss das Let's Encrypt-Zertifikat nach Anleitung generiert werden
- alle SSL-Direktiven in der Konfiguration müssen wieder entkommentiert werden
- ***systemctl reload nginx***

paste.bytespeicher.org

- Datenbank: bs_paste
- Config: /var/www/paste.bytespeicher.org/classes/Config.php

/etc/nginx/sites-available/paste.bytespeicher.org

```
server {
  listen      80;
  listen [::]:80;
  listen      443 ssl;
  listen [::]:443 ssl;

  include snippets/letsencrypt.conf;

  server_name paste.bytespeicher.org;

  if ($scheme != "https") {
    rewrite ^ https://$host$uri permanent;
  }

  ssl on;

  add_header Strict-Transport-Security "max-age=31536000";

  ssl_certificate
/home/letsencrypt/letsencrypt.sh/certs/paste.bytespeicher.org/fullchain.pem;
  ssl_certificate_key
/home/letsencrypt/letsencrypt.sh/certs/paste.bytespeicher.org/privkey.pem;
```

```
  ssl_dhparam /etc/ssl/paste.bytespeicher.org/dhparam.pem;

  ssl_trusted_certificate
/home/letsencrypt/letsencrypt.sh/certs/paste.bytespeicher.org/fullchain.pem;

  root /var/www/paste.bytespeicher.org/;

  index index.php;

  location / {
    try_files $uri $uri/ index.php;
    if ( !-e $request_filename ) {
        rewrite ^/(.*)$ /index.php;
    }
  }
  location ~ .php$ {
    fastcgi_pass   unix:/var/run/php5-fpm.sock;
    fastcgi_index   index.php;
    fastcgi_param   SCRIPT_FILENAME  /var/www/paste.bytespeicher.org/index.php;
    #fastcgi_param   QUERY_STRING $query_string;
    include         fastcgi_params;
  }
  location ~* ^.+\.(jpg|jpeg|gif|bmp|ico|png|css|js|swf)$ {
    expires 30d;
    access_log off;
  }
}
```

bytespeicher.org

- Datenbank: wp_bs
- Config: /var/www/bytespeicher.org/wp-config.php

/etc/nginx/sites-available/bytespeicher.org

```
server {
 listen 80;
 listen [::]:80;

 server_name www.bytespeicher.org staging.bytespeicher.org bytespeicher.org
radio.bytespeicher.org;

 include snippets/letsencrypt.conf;

 if ($host = "radio.bytespeicher.org") {
        rewrite ^ https://bytespeicher.org/category/radio-bytespeicher/ permanent;
 }
 location / {
        rewrite /lpd https://bytespeicher.org/2015/linux-presentation-day-2015/ permanent;
        rewrite ^(.*)$ https://bytespeicher.org$1 permanent;
 }
}

server {
 listen 443;
 listen [::]:443;

 server_name www.bytespeicher.org;

 ssl on;
```

```
 add_header Strict-Transport-Security "max-age=31536000";
 add_header X-Frame-Options SAMEORIGIN;

 ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/bytespeicher.org/fullchain.pem;
 ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/bytespeicher.org/privkey.pem;
 ssl_dhparam /etc/ssl/bytespeicher.org/bytespeicher.org.pem;

 ssl_trusted_certificate
/home/letsencrypt/letsencrypt.sh/certs/bytespeicher.org/fullchain.pem;

 location / {
        rewrite /lpd https://bytespeicher.org/2015/linux-presentation-day-2015/ permanent;
        rewrite ^(.*)$ https://bytespeicher.org$1 permanent;
 }
}

server {
 listen 443;
 listen [::]:443;

 server_name bytespeicher.org;

 ssl on;

 add_header Strict-Transport-Security "max-age=31536000";
 add_header X-Frame-Options SAMEORIGIN;

 ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/bytespeicher.org/fullchain.pem;
 ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/bytespeicher.org/privkey.pem;
 ssl_dhparam /etc/ssl/bytespeicher.org/bytespeicher.org.pem;

 ssl_trusted_certificate
/home/letsencrypt/letsencrypt.sh/certs/bytespeicher.org/fullchain.pem;

 gzip on;
 gzip_disable "msie6";

 gzip_vary on;
 gzip_proxied any;
 gzip_comp_level 6;
 gzip_buffers 16 8k;
 gzip_http_version 1.1;
 gzip_types text/plain text/css application/json application/x-javascript text/xml
application/xml application/xml+rss text/javascript;

 client_max_body_size 64m;

 location / {
        root   /var/www/bytespeicher.org;  # absolute path to your WordPress installation
        index  index.php index.html index.htm;

        rewrite /lpd https://bytespeicher.org/2015/linux-presentation-day-2015/ permanent;

        # this serves static files that exist without running other rewrite tests
        if (-f $request_filename) {
            expires 30d;
            break;
        }

        # this sends all non-existing file or directory requests to index.php
        if (!-e $request_filename) {
```

```
            rewrite ^(.+)$ /index.php?q=$1 last;
        }
    }

 location /piwik/ {
    proxy_pass http://stats.technikkultur-erfurt.de/;
    proxy_set_header X-Real-IP  $remote_addr;
    proxy_set_header X-Forwarded-For $remote_addr;
    proxy_set_header X-Forwarded-Host stats.technikkultur-erfurt.de;

 }

 location /status/ {
    proxy_pass http://status.bytespeicher.org/;
 }

 location ~ .php$ {
  root /var/www/bytespeicher.org;
  fastcgi_keep_conn off;
  fastcgi_pass   unix:/var/run/php5-fpm.sock;
  fastcgi_index  index.php;
  fastcgi_param  SCRIPT_FILENAME  /var/www/bytespeicher.org$fastcgi_script_name;
  include        fastcgi_params;
 }
}
```

status.bytespeicher.org

- ***useradd spacestatus -m -G www-data***
- ***sudo -u spacestatus /bin/bash***
- ***cd ~***
- ***git clone https://github.com/Bytespeicher/space-status// * mkdir www * virtualenv env * . env/bin/activate * pip install jinja2 * crontab -e <file|crontab> # Edit this file to introduce tasks to be run by cron. # # Each task to run has to be defined through a single line # indicating with different fields when the task will be run # and what command to run for the task # # To define the time you can provide concrete values for # minute (m), hour (h), day of month (dom), month (mon), # and day of week (dow) or use '*' in these fields (for 'any').# # Notice that tasks will be started based on the cron's system # daemon's notion of time and timezones. # # Output of the crontab jobs (including errors) is sent through # email to the user the crontab file belongs to (unless redirected). # # For example, you can run a backup of all your user accounts # at 5 a.m every week with: # 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/ # # For more information see the manual pages of crontab(5) and cron(8) # # m h dom mon dow command * * * * * /home/spacestatus/space-status/generate_status 1>/dev/null 2>&1 * * * * * /home/spacestatus/space-status/generate_status_html 1>/dev/null 2>&1 </file> <file|/etc/nginx/sites-available/status.bytespeicher.org> server { listen 80; listen [::]:80; listen 443 ssl; listen [::]:443 ssl; include snippets/letsencrypt.conf; root /home/spacestatus/www; index index.html; server_name status.bytespeicher.org; if ($scheme != „https“) { rewrite ^ https:$host$uri permanent; } location / { try_files $uri $uri/ =404; } ssl on; add_header Strict-Transport-Security „max-age=31536000“; add_header X-Frame-Options SAMEORIGIN; ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/status.bytespeicher.org/fullchain.pem; ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/status.bytespeicher.org/privkey.pem; ssl_dhparam /etc/ssl/status.bytespeicher.org/dhparam.pem; ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/status.bytespeicher.org/fullchain.pem; } </file> ==== makerspace-erfurt.de / fablab-erfurt.de ==== * Datenbank: makerspace_wp * Config: /var/www/makerspace-erfurt.de/public_html/wp-config.php <file|/etc/nginx/sites-available/makerspace-erfurt.de> server { listen 80; listen [::]:80; listen 443; listen [::]:443; server_name makerspace-erfurt.de www.makerspace-erfurt.de fablab-erfurt.de www.fablab-erfurt.de; include snippets/letsencrypt.conf; if ($host != „makerspace-erfurt.de“) { rewrite ^ https://makerspace-erfurt.de$uri permanent; } if ($scheme != „https“) { rewrite ^(.*)$ https://makerspace-erfurt.de$1 permanent; } ssl on; add_header Strict-Transport-Security „max-age=31536000“; add_header X-Frame-Options SAMEORIGIN; ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/makerspace-erfurt.de/fullchain.pem; ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/makerspace-erfurt.de/privkey.pem;***

ssl_dhparam /etc/ssl/makerspace-erfurt.de/dhparam.pem; ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/makerspace-erfurt.de/fullchain.pem; gzip on; gzip_disable „msie6“; gzip_vary on; gzip_proxied any; gzip_comp_level 6; gzip_buffers 16 8k; gzip_http_version 1.1; gzip_types text/plain text/css application/json application/x-javascript text/xml application/xml application/xml+rss text/javascript; client_max_body_size 64m; location / { root /var/www/makerspace-erfurt.de/public_html; # absolute path to your WordPress installation index index.php index.html index.htm; # this serves static files that exist without running other rewrite tests if (-f $request_filename) { expires 30d; break; } # this sends all non-existing file or directory requests to index.php if (!-e $request_filename) { rewrite ^(.+)$ /index.php?q=$1 last; } } location ~ .php$ { root /var/www/makerspace-erfurt.de/public_html; fastcgi_keep_conn off; fastcgi_pass unix:/var/run/php5-fpm.sock; fastcgi_index index.php; fastcgi_param SCRIPT_FILENAME /var/www/makerspace-erfurt.de/public_html/$fastcgi_script_name; include fastcgi_params; } </file> ==== cloud.technikkultur-erfurt.de (Owncloud) ==== * Datenbank: makerspace_oc * Config: /var/www/oc.makerspace-erfurt.de/public_html/config/config.php <file|/etc/nginx/sites-available/cloud.technikkultur-erfurt.de> server { listen 80; listen [::]:80; listen 443 ssl; listen [::]:443 ssl; server_name cloud.technikkultur-erfurt.de oc.makerspace-erfurt.de; include snippets/letsencrypt.conf; if ($scheme != „https“) { return 301 https:$host$request_uri; } ssl on; ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/cloud.technikkultur-erfurt.de/fullchain.pem; ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/cloud.technikkultur-erfurt.de/privkey.pem; ssl_dhparam /etc/ssl/cloud.technikkultur-erfurt.de/dhparam.pem; ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/cloud.technikkultur-erfurt.de/fullchain.pem; # Add headers to serve security related headers # Before enabling Strict-Transport-Security headers please read into this topic first. add_header Strict-Transport-Security „max-age=15552000; includeSubDomains“; add_header X-Content-Type-Options nosniff; add_header X-Frame-Options „SAMEORIGIN“; add_header X-XSS-Protection „1; mode=block“; add_header X-Robots-Tag none; add_header X-Download-Options noopen; add_header X-Permitted-Cross-Domain-Policies none; # The following 2 rules are only needed for the user_webfinger app. # Uncomment it if you're planning to use this app. #rewrite ^/.well-known/host-meta /public.php?service=host-meta last; #rewrite ^/.well-known/host-meta.json /public.php?service=host-meta-json last; location = /.well-known/carddav { return 301 $scheme:$host/remote.php/dav; } location = /.well-known/caldav { return 301 $scheme:$host/remote.php/dav; } root /var/www/oc.makerspace-erfurt.de/public_html/; index index.php; # set max upload size client_max_body_size 512M; fastcgi_buffers 64 4K; # Disable gzip to avoid the removal of the ETag header gzip off; # Uncomment if your server is build with the ngx_pagespeed module # This module is currently not supported. #pagespeed off; error_page 403 /core/templates/403.php; error_page 404 /core/templates/404.php; location / { rewrite ^ /index.php$uri; } location ~ ^/(?:build|tests|config|lib|3rdparty|templates|data)/ { return 404; } location ~ ^/(?:\.|autotest|occ|issue|indie|db_|console) { return 404; } location ~ ^/(?:index|remote|public|cron|core/ajax/update|status|ocs/v[12]|updater/.+|ocs-provider/.+|core/templates/40[34])\.php(?:$|/) { fastcgi_split_path_info ^(.+\.php)(/.*)$; include fastcgi_params; fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name; fastcgi_param PATH_INFO $fastcgi_path_info; fastcgi_param HTTPS on; fastcgi_param modHeadersAvailable true; #Avoid sending the security headers twice fastcgi_param front_controller_active true; fastcgi_pass unix:/var/run/php5-fpm.sock; fastcgi_index index.php; fastcgi_intercept_errors on; #fastcgi_request_buffering off; fastcgi_keep_conn off; include fastcgi_params; } location ~ ^/(?:updater|ocs-provider)(?:$|/) { try_files $uri $uri/ =404; index index.php; } # Adding the cache control header for js and css files # Make sure it is BELOW the PHP block location ~* \.(?:css|js)$ { try_files $uri /index.php$uriis_argsargs; add_header Cache-Control „public, max-age=7200“; # Add headers to serve security related headers (It is intended to have those duplicated to the ones above) # Before enabling Strict-Transport-Security headers please read into this topic first. add_header Strict-Transport-Security „max-age=15552000; includeSubDomains“; add_header X-Content-Type-Options nosniff; add_header X-Frame-Options „SAMEORIGIN“; add_header X-XSS-Protection „1; mode=block“; add_header X-Robots-Tag none; add_header X-Download-Options noopen; add_header X-Permitted-Cross-Domain-Policies none; # Optional: Don't log access to assets access_log off; } location ~* \.(?:svg|gif|png|html|ttf|woff|ico|jpg|jpeg)$ { try_files $uri /index.php$uriis_argsargs; # Optional: Don't log access to other assets access_log off; } location = /robots.txt { allow all; log_not_found off; access_log off; } #access_log /var/log/nginx/oc.makerspace-erfurt.de-access.log; # error_log /var/log/nginx/oc.makerspace-erfurt.de-error.log; } </file> ==== Redmine ==== * Datenbank: redmine Pakete: * thin * ruby * rake * rubygems * ruby-mysql2 * ruby-dev * libmysqlclient-dev * curl * rails * ruby-sass * ruby-compass Installation: <file|/etc/tmpfiles.d/redmine.conf> D /run/thin 0755 redmine redmine - </file> <file|/etc/thin/redmine.yml> — chdir: /home/redmine/redmine environment: production timeout: 30 log: /var/log/thin/redmine.log pid: /var/run/thin/redmine.pid max_conns: 1024 max_persistent_conns:

512 require: [] wait: 30 socket: /var/run/thin/redmine.sock daemonize: true user: redmine group: redmine servers: 1 prefix: / </file> <file|/etc/systemd/system/redmine.service> [Unit] Description=A fast and very simple Ruby web server After=syslog.target network.target [Service] Type=forking User=redmine Group=redmine Environment=„GEM_HOME=~/redmine/vendor/bundle/“ WorkingDirectory=/home/redmine/redmine ExecStart=/usr/bin/bundle exec thin start –config /etc/thin/redmine.yml ExecReload=/usr/bin/bundle exec thin restart –config /etc/thin/redmine.yml ExecStop=/usr/bin/bundle exec thin stop –config /etc/thin/redmine.yml [Install] WantedBy=multi-user.target </file> * mkdir ~/redmine * cd ~/redmine * Redmine-Archiv auspacken * export GEM_HOME='~/redmine/vendor/bundle/' * cp ~/redmine/config/configuration.yml.example ~/redmine/config/configuration.yml * cp ~/redmine/config/database.yml.example ~/redmine/config/database.yml <file|~/redmine/config/database.yml> … production: adapter: mysql2 database: redmine host: localhost username: redmine password: „XXXX“ encoding: utf8 … </file> <file|~/redmine/config/configuration.yml> … production: email_delivery: delivery_method: :smtp smtp_settings: address: mail.bytespeicher.org port: 587 authentication: :plain user_name: 'XXXX' password: 'XXXX' … </file> * bundle install –without development test rmagick * bundle exec rake generate_secret_token * bundle exec rake db:migrate RAILS_ENV=„production“ * RAILS_ENV=production REDMINE_LANG=de bundle exec rake redmine:load_default_data * mkdir /run/thin * chmod 755 /run/thin * chown redmine:redmine /run/thin * systemctl enable redmine.service * systemctl start redmine.service <file|/etc/nginx/sites-available/redmine.bytespeicher.org> server { listen 80; listen [::]:80; listen 443 ssl; listen [::]:443 ssl; include snippets/letsencrypt.conf; server_name redmine.bytespeicher.org; if ($scheme != „https“) { rewrite ^ https:$host$uri permanent; } ssl on; add_header Strict-Transport-Security „max-age=31536000“; ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/redmine.bytespeicher.org/fullchain.pem; ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/redmine.bytespeicher.org/privkey.pem; ssl_dhparam /etc/ssl/redmine.bytespeicher.org/dhparam.pem; ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/redmine.bytespeicher.org/fullchain.pem; root /home/redmine/redmine/public; client_max_body_size 20m; try_files $uri/index.html $uri.html $uri @app; location @app { include /etc/nginx/proxy_params; proxy_pass http://unix:/run/thin/redmine.0.sock; proxy_redirect off; } error_page 500 502 503 504 /500.html; error_page 404 /404.html; } </file> ==== Dokuwiki ==== * DocumentRoot: /var/www/technikkultur-erfurt.de/public_html * Datenverzeichnis: /var/www/technikkultur-erfurt.de/data <file|/etc/nginx/sites-available/technikkultur-erfurt.de> server { listen 80; listen [::]:80; listen 443 ssl; listen [::]:443 ssl; include snippets/letsencrypt.conf; server_name technikkultur-erfurt.de www.technikkultur-erfurt.de; if ($host = „www.technikkultur-erfurt.de“) { rewrite ^ https://technikkultur-erfurt.de$uri permanent; } if ($scheme != „https“) { rewrite ^ https:$host$uri permanent; } ssl on; add_header Strict-Transport-Security „max-age=31536000“; ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/example.org/fullchain.pem; ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/example.org/privkey.pem; ssl_dhparam /etc/ssl/example.org/dhparam.pem; ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/example.org/fullchain.pem; # Maximum file upload size is 4MB - change accordingly if needed client_max_body_size 4M; client_body_buffer_size 128k; root /var/www/technikkultur-erfurt.de/public_html; index doku.php; #Remember to comment the below out when you're installing, and uncomment it when done. location ~ /(data/|conf/|bin/|inc/|install.php) { deny all; } location / { try_files $uri $uri/ @dokuwiki; } location @dokuwiki { rewrite ^/_media/(.*) /lib/exe/fetch.php?media=$1 last; rewrite ^/_detail/(.*) /lib/exe/detail.php?media=$1 last; rewrite ^/_export/([^/]+)/(.*) /doku.php?do=export_$1&id=$2 last; rewrite ^/(.*) /doku.php?id=$1&$args last; } location ~ \.php$ { fastcgi_pass unix:/var/run/php5-fpm.sock; include fastcgi_params; fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name; fastcgi_param REDIRECT_STATUS 200; } } </file> ==== Pad ==== * Software: Etherpad-lite * Datenbank: etherpad-lite Pakete: * nodejs * npm Plugins: * ep_pad-lister Installation: <file|/etc/systemd/system/etherpad-lite.service> [Unit] Description=etherpad-lite (real-time collaborative document editing) After=syslog.target network.target [Service] Type=simple User=etherpad Group=etherpad ExecStart=/home/etherpad/etherpad/bin/run.sh [Install] WantedBy=multi-user.target </file> <file|/etc/nginx/sites-enabled/pad.technikkultur-erfurt.de> server { listen 80; listen [::]:80; listen 443 ssl; listen [::]:443 ssl; server_name pad.technikkultur-erfurt.de; if ($scheme != „https“) { rewrite ^ https:$host$uri permanent; } ssl on; add_header Strict-Transport-Security „max-age=31536000“; ssl_certificate /etc/ssl/pad.technikkultur-erfurt.de/pad.technikkultur-erfurt.de.pem; ssl_certificate_key /etc/ssl/pad.technikkultur-erfurt.de/pad.technikkultur-erfurt.de.key; ssl_dhparam /etc/ssl/pad.technikkultur-erfurt.de/dhparam.pem; ssl_trusted_certificate /etc/ssl/pad.technikkultur-erfurt.de/pad.technikkultur-erfurt.de.pem; location / { include /etc/nginx/proxy_params; proxy_pass http://localhost:13378/; proxy_set_header Host $host; proxy_pass_header Server; # be carefull, this line doesn't override any proxy_buffering on set in a conf.d/file.conf proxy_buffering off; proxy_http_version 1.1; # recommended

*with keepalive connections # WebSocket proxying - from http://nginx.org/en/docs/http/websocket.html proxy_set_header Upgrade $http_upgrade; proxy_set_header Connection $connection_upgrade; } } map $http_upgrade $connection_upgrade { default upgrade; close; } </file> Das Start-Skript für etherpad-lite sucht nach „node“ als nodejs-Server-Binary. Unter Debian lautet es nodejs: * cd /usr/bin/ * ln -s nodejs node Plugin-Installation * sudo -u etherpad /bin/bash * cd ~/etherpad/ * npm install ep_pad-lister Konfiguration <file|~/etherpad/settings.json> { … IP and port which etherpad should bind at „ip“: „127.0.0.1“, „port“ : 13378, … … „dbType“ : „mysql“, „dbSettings“ : { „user“ : „etherpad-lite“, „host“ : „localhost“, „password“: „XXX“, „database“: „etherpad-lite“ }, … } </file> * systemctl enable etherpad-lite.service * systemctl start etherpad-lite.service Migration dirty.db zu MySQL: * https://github.com/ether/etherpad-lite/wiki/Manipulating-the-database ==== wall.technikkultur-erfurt.de ==== * Config: /var/www/wall.technikkultur-erfurt.de/config.php <file|/etc/nginx/sites-available/wall.technikkultur-erfurt.de> server { listen 80; listen [::]:80; server_name wall.technikkultur-erfurt.de; root /var/www/wall.technikkultur-erfurt.de/; index index.php; location ~ .php$ { fastcgi_pass unix:/var/run/php5-fpm.sock; include fastcgi_params; fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name; fastcgi_param REDIRECT_STATUS 200; } } </file> ==== Piwik ==== * Datenbank: bs_piwik * Config: /var/www/stats.technikkultur-erfurt.de/config/config.ini.php <file|/etc/nginx/sites-available/stats.technikkultur-erfurt.de> server { listen 80; listen [::]:80; server_name stats.technikkultur-erfurt.de; root /var/www/stats.technikkultur-erfurt.de/; index index.php; location ~ .php$ { fastcgi_pass unix:/var/run/php5-fpm.sock; include fastcgi_params; fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name; fastcgi_param REDIRECT_STATUS 200; } } </file> ==== Roundcube ==== * Datenbank: roundcubemail * Config: /var/www/mail.bytespeicher.org/config/config.inc.php * mkdir /var/www/mail.bytespeicher.org/ * cd /var/www/mail.bytespeicher.org/ * wget -O /tmp/roundcube.tar.gz https://downloads.sourceforge.net/project/roundcubemail/roundcubemail/1.1.3/roundcubemail-1.1.3-complete.tar.gz * tar -C /var/www/mail.bytespeicher.org/ –strip 1 -tf /tmp/roundcubemail-1.1.3-complete.tar.gz * curl -sS https://getcomposer.org/installer | php * mv composer.json{-dist,} * php composer.phar install –no-dev * chown www-data.www-data -R /var/www/mail.bytespeicher.org * mysql $> CREATE DATABASE roundcubemail; * mysql $> GRANT ALL PRIVILEGES ON roundcubemail.* TO roundcubemail@localhost IDENTIFIED BY '$$password$$'; * mysql $> FLUSH PRIVILEGES; <file|config/config.inc.php> […] $config['db_dsnw'] = 'mysql:roundcubemail:$$password$$/roundcubemail'; $config['default_host'] = array('bytespeicher.org', 'technikkultur-erfurt.de'); $config['product_name'] = 'Bytespeicher Webmail'; $config['des_key'] = '$$random-24-char-des-key$$„; $config['plugins'] = array('archive', 'zipdownload', 'managesieve', 'additional_message_headers', 'attachment_reminder', 'emoticons', 'hide_blockquote', 'jqueryui', 'markasjunk', 'newmail_notifier', 'show_additional_headers', 'subscriptions_option', 'userinfo'); </file> <file|/etc/nginx/sites-available/mail.bytespeicher.org> server { listen 80; listen [::]:80; listen 443 ssl; listen [::]:443 ssl; include snippets/letsencrypt.conf; server_name mail.bytespeicher.org; if ($scheme != „https“) { rewrite ^ https:$host$uri permanent; } ssl on; add_header Strict-Transport-Security „max-age=31536000“; add_header X-Frame-Options SAMEORIGIN; ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/mail.bytespeicher.org/fullchain.pem; ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/mail.bytespeicher.org/privkey.pem; ssl_dhparam /etc/ssl/mail.bytespeicher.org/dhparam.pem; ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/mail.bytespeicher.org/fullchain.pem; root /var/www/mail.bytespeicher.org/; index index.php index.html; location ~ ^/favicon.ico$ { root /var/www/mail.bytespeicher.org/skins/default/images; log_not_found off; access_log off; expires max; } location = /robots.txt { allow all; log_not_found off; access_log off; } location ~ ^/(README|INSTALL|LICENSE|CHANGELOG|UPGRADING)$ { deny all; } location ~ ^/(bin|SQL)/ { deny all; } location ~ /\. { deny all; access_log off; log_not_found off; } location ~ \.php$ { try_files $uri =404; include /etc/nginx/fastcgi_params; fastcgi_pass unix:/var/run/php5-fpm.sock; fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name; fastcgi_index index.php; } location ~* ^.+\.(jpg|jpeg|gif|bmp|ico|png|css|js|swf)$ { expires 30d; access_log off; } } </file> * → Browser → https://mail.bytespeicher.org/install * rm -rf*

*/var/www/mail.bytespeicher.org/installer/ ==== Matrix/Synapse ==== * useradd -m synapse * apt-get install build-essential python2.7-dev libffi-dev python-pip python-setuptools sqlite3 libssl-dev python-virtualenv libjpeg-dev libxslt1-dev coturn * mkdir /home/synapse/ssl * chown synapse:synapse /home/synapse/ssl * chmod 770 /home/synapse/ssl * usermod -G synapse letsencrypt <file|/etc/nginx/sites-enabled/erfurt.chat> server { listen 80; listen [::]:80; listen 443 ssl; listen [::]:443 ssl; server_name erfurt.chat www.erfurt.chat; include snippets/letsencrypt.conf; if ($scheme != „https“) { rewrite ^ https:$host$uri permanent; } if ($host = „www.erfurt.chat“) { rewrite ^ https://erfurt.chat$uri permanent; } root /var/www/erfurt.chat; client_max_body_size 32m; location /_matrix { proxy_pass http://127.0.0.1:8008; proxy_set_header X-Forwarded-For $remote_addr; } ssl on; # add_header Strict-Transport-Security „max-age=31536000“; add_header X-Frame-Options SAMEORIGIN; ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/erfurt.chat/fullchain.pem; ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/erfurt.chat/privkey.pem; ssl_dhparam /etc/ssl/erfurt.chat/dhparam.pem; ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/erfurt.chat/fullchain.pem; access_log /var/log/nginx/erfurt.chat-access.log; error_log /var/log/nginx/erfurt.chat-error.log; } </file> <file|/etc/default/coturn> TURNSERVER_ENABLED=1 </file> <file|/etc/turnserver.conf> external-ip=88.198.111.196 min-port=49152 max-port=59999 lt-cred-mech use-auth-secret static-auth-secret=[your secret key here] realm=erfurt.chat no-tcp no-tls no-tcp-relay cert=/home/letsencrypt/letsencrypt.sh/certs/erfurt.chat/cert.pem pkey=/home/letsencrypt/letsencrypt.sh/certs/erfurt.chat/privkey.pem cipher-list=„EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH:!AES128“ syslog denied-peer-ip=10.0.0.0-10.255.255.255 denied-peer-ip=192.168.0.0-192.168.255.255 denied-peer-ip=172.16.0.0-172.31.255.255 allowed-peer-ip=172.31.1.100 no-sslv2 no-sslv3 </file> * service coturn restart * sudo -u synapse /bin/bash * cd * virtualenv -p python2.7 ~/.synapse * source ~/.synapse/bin/activate * pip install –upgrade pip * pip install –upgrade setuptools * pip install lxml * pip install https://github.com/matrix-org/synapse/tarball/master * cd ~/.synapse * python -m synapse.app.homeserver –server-name erfurt.chat –config-path homeserver.yaml –generate-config –report-stats=no <file|/home/synapse/.synapse/homeserver.yaml> –homeserver.yaml.orig 2017-06-05 12:56:46.729514635 +0200 +++ homeserver.yaml 2017-06-05 18:44:13.546761068 +0200 @@ -4,10 +4,10 @@ # autogenerates on launch with your own SSL certificate + key pair # if you like. Any required intermediary certificates can be # appended after the primary certificate in hierarchical order. -tls_certificate_path: “/home/synapse/.synapse/erfurt.chat.tls.crt„ +tls_certificate_path: “/home/synapse/ssl/fullchain.pem„ # PEM encoded private key for TLS -tls_private_key_path: “/home/synapse/.synapse/erfurt.chat.tls.key„ +tls_private_key_path: “/home/synapse/ssl/privkey.pem„ # PEM dh parameters for ephemeral keys tls_dh_params_path: “/home/synapse/.synapse/erfurt.chat.tls.dh„ @@ -50,7 +50,7 @@ pid_file: /home/synapse/.synapse/homeserver.pid # Whether to serve a web client from the HTTP/HTTPS root resource. -web_client: True +web_client: False # The root directory to server for the above web client. # If left undefined, synapse will serve the matrix-angular-sdk web client. @@ -59,7 +59,7 @@ # web_client_location: “/path/to/web/root„ # The public-facing base URL for the client API (not including _matrix/…) -# public_baseurl: https://example.com:8448/ +public_baseurl: https://erfurt.chat:8448/ # Set the soft limit on the number of file descriptors synapse can use # Zero is used to indicate synapse should set the soft limit to the @@ -123,7 +123,7 @@ bind_addresses: ['0.0.0.0'] type: http - x_forwarded: false + x_forwarded: True resources: - names: [client, webclient] @@ -231,7 +231,7 @@ # Is the preview URL API enabled? If enabled, you *must* specify # an explicit url_preview_ip_range_blacklist of IPs that the spider is # denied from accessing. -url_preview_enabled: False +url_preview_enabled: True # List of IP address CIDR ranges that the URL preview spider is denied # from accessing. There are no defaults: you must explicitly @@ -241,14 +241,14 @@ # synapse to issue arbitrary GET requests to your internal services, # causing serious security issues. # -# url_preview_ip_range_blacklist: -# - '127.0.0.0/8' -# - '10.0.0.0/8' -# - '172.16.0.0/12' -# - '192.168.0.0/16' -# - '100.64.0.0/10' -# - '169.254.0.0/16' -# +url_preview_ip_range_blacklist: + - '127.0.0.0/8' + - '10.0.0.0/8' + - '172.16.0.0/12' + - '192.168.0.0/16' + - '100.64.0.0/10' + - '169.254.0.0/16' + # List of IP address*

*CIDR ranges that the URL preview spider is allowed # to access even if they are specified in url_preview_ip_range_blacklist. # This is useful for specifying exceptions to wide-ranging blacklisted @@ -322,10 +322,10 @@ ## Turn ## # The public URIs of the TURN server to give to clients -turn_uris: [] +turn_uris: [„turn:erfurt.chat:3478?transport=udp“, „turn:erfurt.chat:3478?transport=tcp“] # The shared secret used to compute passwords for the TURN server -turn_shared_secret: „YOUR_SHARED_SECRET“ +turn_shared_secret: „$$$SECRET$$$“ # The Username and password if the TURN server needs them and # does not use a token @@ -346,7 +346,7 @@ ## Registration ## # Enable registration for new users. -enable_registration: False +enable_registration: True # If set, allows registration by anyone who also has the shared # secret, even if registration is otherwise disabled. @@ -360,7 +360,7 @@ # Allows users to register as guests without a password/email/etc, and # participate in rooms hosted on this server which have been made # accessible to anonymous users. -allow_guest_access: False +allow_guest_access: True # The list of identity servers trusted to verify third party # identifiers by this server. @@ -388,7 +388,7 @@ # A list of application service config file to use -app_service_config_files: [] +app_service_config_files: [„ircbridge_registration.yaml“] macaroon_secret_key: „$$$SECRET$$$“ @@ -461,7 +461,8 @@ enabled: true # Uncomment and change to a secret random string for extra security. # DO NOT CHANGE THIS AFTER INITIAL SETUP! - #pepper: “„ + pepper: „$$$SECRET$$$“ + @@ -473,20 +474,20 @@ # If your SMTP server requires authentication, the optional smtp_user & # smtp_pass variables should be used # -#email: -# enable_notifs: false -# smtp_host: „localhost“ -# smtp_port: 25 -# smtp_user: „exampleusername“ -# smtp_pass: „examplepassword“ -# require_transport_security: False -# notif_from: „Your Friendly %(app)s Home Server noreply@example.com“ -# app_name: Matrix -# template_dir: res/templates -# notif_template_html: notif_mail.html -# notif_template_text: notif_mail.txt -# notif_for_new_users: True -# riot_base_url: „http://localhost/riot“ +email: + enable_notifs: True + smtp_host: „localhost“ + smtp_port: 587 + smtp_user: „synapse@erfurt.chat“ + smtp_pass: „$$$SECRET$$$“ + require_transport_security: True + notif_from: „Your Friendly %(app)s Home Server noreply@erfurt.chat“ + app_name: Matrix + template_dir: res/templates + notif_template_html: notif_mail.html + notif_template_text: notif_mail.txt + notif_for_new_users: True + riot_base_url: „https://erfurt.chat/riot“ # password_providers: </file> <file|/etc/systemd/system/synapse.service> [Unit] Description=Synapse Matrix homeserver [Service] Type=simple User=synapse Group=synapse #EnvironmentFile=-/etc/sysconfig/synapse WorkingDirectory=/home/synapse/.synapse ExecStart=/home/synapse/.synapse/bin/python2.7 -m synapse.app.homeserver —config-path=/home/synapse/.synapse/homeserver.yaml [Install] WantedBy=multi-user.target </file> * systemctl enable synapse * systemctl start synapse * wget -O /usr/src/vector-im-v0.10.1.tar.gz https://github.com/vector-im/riot-web/releases/download/v0.10.1/riot-v0.10.1.tar.gz * mkdir /var/www/erfurt.chat/ * tar —strip-components=1 -xf /usr/src/vector-im-v0.10.1.tar.gz -C /var/www/erfurt.chat/ <file|/var/www/erfurt.chat/config.json> { „default_hs_url“: „https://erfurt.chat“, „default_is_url“: „https://vector.im“, „brand“: „erfurt.chat“, „integrations_ui_url“: „https://scalar.vector.im/“, „integrations_rest_url“: „https://scalar.vector.im/api“, „bug_report_endpoint_url“: „https://riot.im/bugreports/submit“, „enableLabs“: true, „roomDirectory“: { „servers“: [„erfurt.chat“, „matrix.org“] }, } </file> === Matrix IRC Bridge === * curl -sL https://deb.nodesource.com/setup_6.x | sudo -E bash - * apt-get install -y nodejs * npm install matrix-appservice-irc —global <file|/home/synapse/.synapse/ircbridge_config.yaml> homeserver: url: „https://erfurt.chat“ # CAUTION: This is a very coarse heuristic. Federated homeservers may have different # clock times and hence produce different origin_server_ts values, which may be old # enough to cause *all* events from the homeserver to be dropped. # Default: 0 (don't ever drop) # dropMatrixMessagesAfterSecs: 300 # 5 minutes domain: „erfurt.chat“ ircService: servers: „irc.hackint.org“: name: „Hackint“ networkId: „hackint“ port: 9999 ssl: true sslselfsign: true ca: | ——BEGIN CERTIFICATE——*

*MIIGBzCCA++gAwIBAgIJAKZfNgKecw1WMA0GCSqGSIb3DQEBCwUAMIGEMRwwGgYD
VQQKExNIYWNraW50IElSQyBOZXR3b3JrMR8wHQYDVQQLExZodHRwOi8vd3d3Lmhh
Y2tpbnQub3JnMSQwIgYDVQQDExtIYWNraW50IElSQyBOZXR3b3JrIFJvb3QgQ0Ex
HTAbBgkqhkiG9w0BCQEWDmNhQGhhY2tpbnQub3JnMB4XDTE1MDcwMTAwMDAwMFoX
DTM1MTIzMTIzNTk1OVowgYQxHDAaBgNVBAoTE0hhY2tpbnQgSVJDIE5ldHdvcmsx*

*HzAdBgNVBAsTFmh0dHA6Ly93d3cuaGFja2ludC5vcmcxJDAiBgNVBAMTG0hhY2tp
bnQgSVJDIE5ldHdvcmsgUm9vdCBDQTEdMBsGCSqGSIb3DQEJARYOY2FAaGFja2lu
dC5vcmcwggIiMA0GCSqGSIb3DQEBAQUAA4ICDwAwggIKAoICAQDi57PWGLHMfxlN
yjtXUS4oYK77+C1ByJtziDWYbEiamrmYbOZ3ukzfH4nHHOLuAiQIT8Tw8gVXMw6w
CNplAUN0mAIQhhu10PwsBLjf638F/NTPzBmziMZyyuSrvyAkZp6Ktv5DAXymIV6C
7LmVwhJiqC5+YFC1JbZJt8wGrew/YLrroYUJm0n7FpW/EkUrl3cQOHIV5xFl9LxR
4xh/lC1AuAsawv8vaxQFGiun25F4jd6l/Evf0tr628kpEXH4hspkeNsQh9uUUpjx
CpNQqh7Wyi1M/QhiK9GFuODd0wsU77iOfccJl3FVf/bjLcO9COMLOBWaJgEpJMNw
j2uBk7pMKScw3S2qvtqBxf7VtfvlyPeX5C7+XCXXViBFcYubzmNlNq5n3qEbMG4t
qwdxR4Mhbhy4BhOGkFNdURsf4N47TvPV6eglHPLc05uYvL5VIddNxH1jrpxVYX76
KXvpR4+vUTYYVi8m2A4Rf+JMI5CELfie2chghhiojAuKDuKfmW3v+fkuGkjEC5A8
NfzD7EOGJB2osAbKP6rx77tVuAo0eMPLHijpgYciXIGoprwqFrjttvRaMkGywwLq
6JDyfB8hMvMvmVPnqx4zbmOaS/Ut2irVwU0k9jiDN29dTvc3ySHwW0bd+Lt5fWJD
DL/lb2it8Z8pJYmZwt1e7vl4LNdm2QIDAQABo3oweDAPBgNVHRMBAf8EBTADAQH/
MB0GA1UdDgQWBBQVmc++GVicHQ7I4FDpPkZdr3nNCzAOBgNVHQ8BAf8EBAMCAQYw
NgYIKwYBBQUHAQEEKjAoMCYGCCsGAQUFBzAChhpodHRwOi8vaGFja2ludC5vcmcv
Y2EuaHRtbDANBgkqhkiG9w0BAQsFAAOCAgEAG82hdmLpfvG7RYbtCb6F4u8FBFxv
zR4Ye5nOPBKaA+CHA+KGScnBFg/E+aMI+IQ3j4Sgar0MZKwu5fI3ETdYReXWtSuE
3/UnT9U1ffUTTNuKwkFM3p5byrVzgmF3fI7aSAFyoa88xl6R/fzjXrXCp+eCy/tE
LTma2WRh+VORCX397h+FFVux3JtfBD+6uW53MOmNvSd2hndi8RpVbgklMfUWxcwK
z+R97QXhNopH33J1rmRm9/RUadKjChiIe+zM/eZJUPObIqiCaCP/qVAxruwHTi8E
tpNFNTCOxe0lwZ6lVNLWun7zY3+vk0Puk6KqnfBlNGK1QDxkTQLILdgGo5WQ11YN
oMmHGztLgZtiWLGLNhTrtAIRNKuc3sw0BOlv+osiH+KvDNvRKufc2eNkaGfLq7TJ
dhiAK2gKkYYAQ5zfDBwSspbtCsszYgEAin3PqoQUdG8f+4I49E0xS7PWQE75e7J9
MCnElQxAPWk9xuZhtkeWUHskpCjrNO7k3dshV0frn2OxPtSgQjjtZxQKQZYzQfPk
j/eVuFwWxQY9pZdOku7fRGbaLEyTbQHZW802rgmaLxxItWQKqZxG1Za7RlKo4Wur
9ZGuYKMAEnPmhJj2KlmXJAaIdQF6LA3NS0KvpWtOfrjaaroHHOUnrxBxCBlfoBpw w3r7JBQGOVK95Sw= ——END
CERTIFICATE—— # The connection password to send for all clients as a PASS command.
Optional. # password: 'pa$$w0rd' sendConnectionMessages: false quitDebounce: # Whether
parts due to net-splits are debounced for delayMs, to allow # time for the netsplit to
resolve itself. A netsplit is detected as being # a QUIT rate higher than
quitsPerSecond. Default: false. enabled: false # The maximum number of quits per second
acceptable above which a netsplit is # considered ongoing. Default: 5. quitsPerSecond: 5
The ti # a net # is not sent many requests to leave rooms all at once if a netsplit
occurs and many # people to not rejoin. # If the user with the same IRC nick as the one
who sent the quit rejoins a channel # they are considered back online and the quit is
not bridged, so long as the rejoin # occurs before the randomly-jittered timeout is not
reached. # Default: 3600000, = 1h delayMinMs: 3600000 # 1h # Default: 7200000, = 2h
delayMaxMs: 7200000 # 2h modePowerMap: o: 50 botConfig: enabled: true nick: „MatrixBot“
password: „$$$$SECRET$$$$“ joinChannelsIfNoUsers: true privateMessages: enabled: true #
exclude: [„Alice“, „Bob“] # NOT YET IMPLEMENTED federate: true # Configuration for
mappings not explicitly listed in the 'mappings' # section. dynamicChannels: # Enable
the ability for Matrix users to join *any* channel on this IRC # network. # Default:
false. enabled: true # Should the AS create a room alias for the new Matrix room? The
form of # the alias can be modified via 'aliasTemplate'. Default: true. createAlias:
true # Should the AS publish the new Matrix room to the public room list so # anyone can
see it? Default: true. published: true # What should the join_rule be for the new Matrix
room? If 'public', # anyone can join the room. If 'invite', only users with an invite
can # join the room. Note that if an IRC channel has +k or +i set on it, # join_rules
will be set to 'invite' until these modes are removed. # Default: „public“. joinRule:
public # Should created Matrix rooms be federated? If false, only users on the # HS
attached to this AS will be able to interact with this room. # Default: true. federate:
true # The room alias template to apply when creating new aliases. This only # applies
if createAlias is 'true'. The following variables are exposed: # $SERVER ⇒ The IRC
server address (e.g. „irc.example.com“) # $CHANNEL ⇒ The IRC channel (e.g. “#python„) #
This MUST have $CHANNEL somewhere in it. # Default: '#irc_$SERVER_$CHANNEL'
#aliasTemplate: “#irc_$CHANNEL„ # A list of user IDs which the AS bot will send invites
to in response # to a !join. Only applies if joinRule is 'invite'. Default: [] #
whitelist: # - “@foo:example.com„ # - “@bar:example.com„ # # Prevent the given list of
channels from being mapped under any # circumstances. # exclude: [“#foo„, “#bar„] #
Configuration for controlling how Matrix and IRC membership lists are # synced.*

membershipLists: # Enable the syncing of membership lists between IRC and Matrix. This # can have a significant effect on performance on startup as the lists are # synced. This must be enabled for anything else in this section to take # effect. Default: false. enabled: true # Syncing membership lists at startup can result in hundreds of members to # process all at once. This timer drip feeds membership entries at the # specified rate. Default: 10000. (10s) floodDelayMs: 10000 global: ircToMatrix: # Get a snapshot of all real IRC users on a channel (via NAMES) and # join their virtual matrix clients to the room. initial: true # Make virtual matrix clients join and leave rooms as their real IRC # counterparts join/part channels. Default: false. incremental: true matrixToIrc: # Get a snapshot of all real Matrix users in the room and join all of # them to the mapped IRC channel on startup. Default: false. initial: true # Make virtual IRC clients join and leave channels as their real Matrix # counterparts join/leave rooms. Make sure your 'maxClients' value is # high enough! Default: false. incremental: true # Apply specific rules to Matrix rooms. Only matrix-to-IRC takes effect. rooms: - room: "!fuasirouddJoxtwfge:localhost„ matrixToIrc: initial: false incremental: false # Apply specific rules to IRC channels. Only IRC-to-matrix takes effect. channels: - channel: "#foo„ ircToMatrix: initial: false incremental: false mappings: # 1:many mappings from IRC channels to room IDs on this IRC server. # The matrix room must already exist. Your matrix client should expose # the room ID in a „settings" page for the room. #"#bytespeicher-testing„: ["„, "!SUxMWVVxsKCFfBsKrR:unikorn.me„] "#bytespeicher„: ["!bGHdpETBTpNZzPzIDo:erfurt.chat„] # Configuration for virtual matrix users. The following variables are # exposed: # $NICK ⇒ The IRC nick # $SERVER ⇒ The IRC server address (e.g. „irc.example.com") matrixClients: # The user ID template to use when creating virtual matrix users. This # MUST have $NICK somewhere in it. # Optional. Default: "@$SERVER_$NICK„. # Example: "@irc.example.com_Alice:example.com„ userTemplate: "@irc_$NICK„ # The display name to use for created matrix clients. This should have # $NICK somewhere in it if it is specified. Can also use $SERVER to # insert the IRC domain. # Optional. Default: „$NICK (IRC)". Example: „Alice (IRC)" displayName: „$NICK (IRC)" # Configuration for virtual IRC users. The following variables are exposed: # $LOCALPART ⇒ The user ID localpart („alice" in @alice:localhost) # $USERID ⇒ The user ID # $DISPLAY ⇒ The display name of this user, with excluded characters # (e.g. space) removed. If the user has no display name, this # falls back to $LOCALPART. ircClients: # The template to apply to every IRC client nick. This MUST have either # $DISPLAY or $USERID or $LOCALPART somewhere in it. # Optional. Default: „M-$DISPLAY". Example: „M-Alice". nickTemplate: „$DISPLAY[m]" # True to allow virtual IRC clients to change their nick on this server # by issuing !nick <server> <nick> commands to the IRC AS bot. # This is completely freeform: it will NOT follow the nickTemplate. allowNickChanges: true # The max number of IRC clients that will connect. If the limit is # reached, the client that spoke the longest time ago will be # disconnected and replaced. # Optional. Default: 30. maxClients: 30 # IPv6 configuration. ipv6: # Optional. Set to true to force IPv6 for outgoing connections. only: false # Optional. The IPv6 prefix to use for generating unique addresses for each # connected user. If not specified, all users will connect from the same # (default) address. This may require additional OS-specific work to allow # for the node process to bind to multiple different source addresses # e.g IP_FREEBIND on Linux, which requires an LD_PRELOAD with the library # https://github.com/matrix-org/freebindfree as Node does not expose setsockopt. prefix: „2a01:4f8:c17:1214::1:" # modify appropriately # # The maximum amount of time in seconds that the client can exist # without sending another message before being disconnected. Use 0 to # not apply an idle timeout. This value is ignored if this IRC server is # mirroring matrix membership lists to IRC. Default: 172800 (48 hours) idleTimeout: 10800 # The number of millseconds to wait between consecutive reconnections if a # client gets disconnected. Setting to 0 will cause the scheduling to be # disabled, i.e. it will be scheduled immediately (with jitter. # Otherwise, the scheduling interval will be used such that one client # reconnect for this server will be handled every reconnectIntervalMs ms using # a FIFO queue. # Default: 5000 (5 seconds) reconnectIntervalMs: 5000 # The number of lines to allow being sent by the IRC client that has received # a large block of text to send from matrix. If the number of lines that would # be sent is > lineLimit, the text will instead be uploaded to matrix and the # resulting URI is treated as a file. As such, a link will be sent to the IRC # side instead of potentially spamming IRC and getting the IRC client kicked. # Default: 3. lineLimit: 3 # A list of user modes to set on every IRC client. For example, „RiG" would

*set # +R, +i and +G on every IRC connection when they have successfully connected. # User modes vary wildly depending on the IRC network you're connecting to, # so check before setting this value. Some modes may not work as intended # through the bridge e.g. caller ID as there is no way to /ACCEPT. # Default: "„ (no user modes) # userModes: „R" # Configuration for an ident server. If you are running a public bridge it is # advised you setup an ident server so IRC mods can ban specific matrix users # rather than the application service itself. ident: # True to listen for Ident requests and respond with the # matrix user's user_id (converted to ASCII, respecting RFC 1413). # Default: false. enabled: false # The port to listen on for incoming ident requests. # Ports below 1024 require root to listen on, and you may not want this to # run as root. Instead, you can get something like an Apache to yank up # incoming requests to 113 to a high numbered port. Set the port to listen # on instead of 113 here. # Default: 113. port: 1113 # Configuration for logging. Optional. Default: console debug level logging # only. logging: # Level to log on console/logfile. One of error|warn|info|debug level: „debug" # The file location to log to. This is relative to the project directory. logfile: „debug.log" # The file location to log errors to. This is relative to the project # directory. errfile: „errors.log" # Whether to log to the console or not. toConsole: true # The max size each file can get to in bytes before a new file is created. maxFileSizeBytes: 134217728 # 128 MB # The max number of files to keep. Files will be overwritten eventually due # to rotations. maxFiles: 5 # Optional. Enable Prometheus metrics. If this is enabled, you MUST install `prom-client`: # $ npm install prom-client@6.3.0 # Metrics will then be available via GET /metrics on the bridge listening port (-p). # metrics: # enabled: true # The nedb database URI to connect to. This is the name of the directory to # dump .db files to. This is relative to the project directory. # Required. databaseUri: „nedb:data" # Configuration options for the debug HTTP API. To access this API, you must # append ?access_token=$APPSERVICE_TOKEN (from the registration file) to the requests. # # The debug API exposes the following endpoints: # # GET /irc/$domain/user/$user_id ⇒ Return internal state for the IRC client for this user ID. # # POST /irc/$domain/user/$user_id ⇒ Issue a raw IRC command down this connection. # Format: new line delimited commands as per IRC protocol. # debugApi: # True to enable the HTTP API endpoint. Default: false. enabled: false # The port to host the HTTP API. port: 11100 # Configuration for the provisioning API. # # GET /_matrix/provision/link # GET /_matrix/provision/unlink # GET /_matrix/provision/listlinks # provisioning: # True to enable the provisioning HTTP endpoint. Default: false. enabled: false # The number of seconds to wait before giving up on getting a response from # an IRC channel operator. If the channel operator does not respond within the # allotted time period, the provisioning request will fail. # Default: 300 seconds (5 mins) requestTimeoutSeconds: 300 # WARNING: The bridge needs to send plaintext passwords to the IRC server, it cannot # send a password hash. As a result, passwords (NOT hashes) are stored encrypted in # the database. # # To generate a .pem file: # $ openssl genpkey -out passkey.pem -outform PEM -algorithm RSA -pkeyopt rsa_keygen_bits:2048 # # The path to the RSA PEM-formatted private key to use when encrypting IRC passwords # for storage in the database. Passwords are stored by using the admin room command # `!storepass server.name passw0rd. When a connection is made to IRC on behalf of # the Matrix user, this password will be sent as the server password (PASS command). passwordEncryptionKeyPath: „passkey.pem" </file> <file|/etc/systemd/system/matrix-irc-bridge.service> [Unit] Description=Matrix IRC Bridge [Service] Type=simple User=synapse Group=synapse #EnvironmentFile=-/etc/sysconfig/synapse WorkingDirectory=/home/synapse/.synapse ExecStart=/usr/local/bin/matrix-appservice-irc -c ircbridge_config.yaml -f ircbridge.yaml -p 9999 [Install] WantedBy=multi-user.target </file> * matrix-appservice-irc -r -f ircbridge_registration.yaml -u „http://erfurt.chat:9999" -c ircbridge_config.yaml -l ircbridge * systemctl enable matrix-irc-bridge.service * systemctl start matrix-irc-bridge.service === Externe Synapse Dokumentation === * https://github.com/matrix-org/synapse/blob/master/README.rst#synapse-installation * https://github.com/matrix-org/synapse/blob/master/README.rst#setting-up-federation * https://github.com/matrix-org/synapse/blob/master/docs/turn-howto.rst ==== users.bytespeicher.org ==== <file|/etc/nginx/sites-available/users.bytespeicher.org> server { listen 80; listen [::]:80; index index.html; server_name users.bytespeicher.org; location / { try_files $uri $uri/ =404; } location ~ ^/~(.+?)(/.*)?$ { alias /home/$1/public_html$2; index index.html index.htm; autoindex*

*on; } } </file> ===== Datensicherung ===== Die Datensicherung erfolgt verschlüsselt auf einen Server von mape2k und einen Server von mkzero: * 1 Full-Backup je Woche * Inkrementelle Backups täglich * Vorhaltezeit: 4 Wochen Pakete: * duply * duplicity * lftp Installation nach folgender Anleitung: https://wiki.fem.tu-ilmenau.de/public/technik/howto/duply * MySQL-Dump-Skript unter /usr/local/bin/mysql-dump einrichten * duply mape2k-backup create Konfiguration: <file|.duply/mape2k-backup/conf> # GPG_KEY='_KEY_ID_' GPG_PW= GPG_KEY_SIGN='58252DC6' GPG_KEYS_ENC='DD379EDC' GPG_PW_SIGN='XXXXXXXXXXXXXXX' TARGET='ftps:XXXXX.YYY.ZZ/' TARGET_USER='bytecluster0001.bytespeicher.org' TARGET_PASS='XXXXXX' # base directory to backup SOURCE='/' MAX_AGE=4W MAX_FULL_BACKUPS=4 MAX_FULLBKP_AGE=1W DUPL_PARAMS=„$DUPL_PARAMS –full-if-older-than $MAX_FULLBKP_AGE “ VOLSIZE=250 DUPL_PARAMS=„$DUPL_PARAMS –volsize $VOLSIZE “ #VERBOSITY=5 </file> <file|.duply/mkzero-backup/conf> # GPG_KEY='_KEY_ID_' GPG_PW='' GPG_KEY_SIGN='58252DC6' GPG_KEYS_ENC='DD379EDC' GPG_PW_SIGN='XXXXXXXXXXXXXXX' TARGET='sftp:XXXXX.YYY.ZZ/' TARGET_USER='bytespeicher' TARGET_PASS='XXXXXX' # base directory to backup SOURCE='/' MAX_AGE=4W MAX_FULL_BACKUPS=4 MAX_FULLBKP_AGE=1W DUPL_PARAMS=„$DUPL_PARAMS –full-if-older-than $MAX_FULLBKP_AGE “ VOLSIZE=250 DUPL_PARAMS=„$DUPL_PARAMS –volsize $VOLSIZE “ #VERBOSITY=5 </file> Verzeichnisausnahmen: <file|.duply/mape2k-backup/exclude> + /tmp/mysqldump - /dev - /sys - /proc - /run - /tmp - /var/tmp - /root/.cache - /root/backup </file> Benutzer für Sicherung der Datenbank einrichten: <file|Benutzer für Datensicherung> CREATE USER 'backup'@'localhost' IDENTIFIED BY 'PASSWORT'; GRANT USAGE ON * . * TO 'backup'@'localhost' IDENTIFIED BY 'PASSWORT' WITH MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0 ; REVOKE ALL PRIVILEGES ON * . * FROM 'backup'@'localhost'; REVOKE GRANT OPTION ON * . * FROM 'backup'@'localhost'; GRANT SELECT, SHOW DATABASES, LOCK TABLES, SHOW VIEW ON * . * TO 'backup'@'localhost' WITH MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0; FLUSH PRIVILEGES; </file> Zusätzliche Sicherung der Datenbanken _vor_ der Datensicherung: <file|.duply/mape2k-backup/pre > mkdir -p /tmp/mysqldump /usr/local/bin/mysql-dump </file> <file|.duply/mape2k-backup/post> /bin/rm -rf /tmp/mysqldump </file> <file|/usr/local/bin/mysql-dump.cnf> [client] user=backup password=„PASSWORT“ host=localhost </file> Sicherung per Cronjob: <file|/etc/crontab> # Backup (mape2k) 0 4 * * 1 root HOME=/root && duply mape2k-backup cleanup_purge_purge-full –extra-clean –force 30 4 * * * root HOME=/root && duply mape2k-backup backup # Backup (mkzero) 0 2 * * 1 root HOME=/root && duply mkzero-backup cleanup_purge_purge-full –extra-clean –force 30 2 * * * root HOME=/root && duply mkzero-backup backup </file>*