

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

Pakete

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

Konfiguration SSH

- HostKey DSA entfernt
- PermitRootLogin no
- PasswordAuthentication no

/etc/motd


```

: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

# 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

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       = 0_DIRECT
+innodb_file_per_table
+
+innodb_thread_concurrency = 8
+
[mysqldump]
quick
quote-names

```

NGINX

- nginx

/etc/nginx/patch

```

diff -Naur /etc/nginx.dist/conf.d/ssl.conf /etc/nginx/conf.d/ssl.conf
--- /etc/nginx.dist/conf.d/ssl.conf 1970-01-01 01:00:00.000000000 +0100
+++ /etc/nginx/conf.d/ssl.conf 2015-11-04 22:41:34.269315957 +0100
@@ -0,0 +1,12 @@
+ssl_ciphers "EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH";
+ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
+ssl_prefer_server_ciphers on;
+ssl_session_cache shared:SSL:10m;
+add_header Strict-Transport-Security "max-age=63072000; includeSubdomains; preload";
+add_header X-Frame-Options DENY;
+add_header X-Content-Type-Options nosniff;
+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
+resolver 213.133.98.98 213.133.99.99 valid=300s;
+resolver_timeout 5s;
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: P00DLE
- ssl_prefer_server_ciphers on;
+ #ssl_protocols TLSv1 TLSv1.1 TLSv1.2; # Dropping SSLv3, ref: P00DLE
+ #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>**
- **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

    # 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;
```

```

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

ssl_prefer_server_ciphers on;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-
GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-
SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-
SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-
SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-
SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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

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_stapling on;
ssl_stapling_verify on;
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-mysqldb
- 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**

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;

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

    ssl_prefer_server_ciphers on;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:CDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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

    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_stapling on;
ssl_stapling_verify on;
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;

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

ssl_prefer_server_ciphers on;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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_stapling on;
ssl_stapling_verify on;
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;

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

    ssl_prefer_server_ciphers on;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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

    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_stapling on;
    ssl_stapling_verify on;
    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;

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

ssl_prefer_server_ciphers on;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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

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_stapling on;
ssl_stapling_verify on;
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***

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

```

/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;

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

    ssl_prefer_server_ciphers on;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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

    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_stapling on;
    ssl_stapling_verify on;
    ssl_trusted_certificate
/home/letsencrypt/letsencrypt.sh/certs/status.bytespeicher.org/fullchain.pem;
}
```

makerspace-erfurt.de / fablab-erfurt.de

- Datenbank: makerspace_wp
- Config: /var/www/makerspace-erfurt.de/public_html/wp-config.php

/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;

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

    ssl_prefer_server_ciphers on;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-
    SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-
    AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-
    SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-
    GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-
    SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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

    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_stapling on;
    ssl_stapling_verify on;
    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;
}

```

oc.makerspace-erfurt.de (Owncloud)

- Datenbank: makerspace_oc
- Config: /var/www/oc.makerspace-erfurt.de/public_html/config/config.php

/etc/nginx/sites-available/oc.makerspace-erfurt.de

```

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

    server_name oc.makerspace-erfurt.de;

    include snippets/letsencrypt.conf;

    if ($scheme != "https") {
        rewrite ^(.*)$ https://oc.makerspace-erfurt.de$1 permanent;
    }

    ssl on;

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

    ssl_prefer_server_ciphers on;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDSA-AES256-GCM-SHA384:ECDSA-AES128-GCM-SHA256:ECDSA-AES256-SHA384:ECDSA-AES128-SHA256:ECDSA-AES256-SHA:ECDSA-AES128-SHA:EDH-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";
}

```

```
ssl_certificate /home/letsencrypt/letsencrypt.sh/certs/oc.makerspace-erfurt.de/fullchain.pem;
ssl_certificate_key /home/letsencrypt/letsencrypt.sh/certs/oc.makerspace-erfurt.de/privkey.pem;
ssl_dhparam /etc/ssl/oc.makerspace-erfurt.de/dhparam.pem;

ssl_stapling on;
ssl_stapling_verify on;
ssl_trusted_certificate /home/letsencrypt/letsencrypt.sh/certs/oc.makerspace-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(?:$|/)) {
    fastcgi_split_path_info ^(.+\.(?:php|php5))(/.*)$;
    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 $uri /index.php$uri$is_args$args;
    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 $uri /index.php$uri$is_args$args;
    # 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;
}
```

Redmine

- Datenbank: redmine

Pakete:

- thin
- ruby
- rake
- rubygems
- ruby-mysql2
- ruby-dev
- libmysqlclient-dev
- curl
- rails
- ruby-sass
- ruby-compass

Installation:

```
/etc/tmpfiles.d/redmine.conf
```

```
D /run/thin 0755 redmine redmine -
```

```
/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: /
```

```
/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
```

- ***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***

```
~/redmine/config/database.yml
```

```
...
production:
  adapter: mysql2
  database: redmine
  host: localhost
  username: redmine
  password: "XXXX"
  encoding: utf8
...
```

```
~/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'
...
```

- ***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***

```
/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;

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

ssl_prefer_server_ciphers on;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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_stapling on;
ssl_stapling_verify on;
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;
}

```

Dokuwiki

- DocumentRoot: /var/www/technikkultur-erfurt.de/public_html
- Datenverzeichnis: /var/www/technikkultur-erfurt.de/data

/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;

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

ssl_prefer_server_ciphers on;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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_stapling on;
ssl_stapling_verify on;
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;
}
}
```

Pad

- Software: Etherpad-lite
- Datenbank: etherpad-lite

Pakete:

- nodejs
- npm

Plugins:

- ep_padlist

Installation:

/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
```

/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;

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

    ssl_prefer_server_ciphers on;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-
```

```

SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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_stapling on;
ssl_stapling_verify on;
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;
}

```

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

~/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"
},
...
}
```

- ***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

/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;
    }
}
```

Piwik

- Datenbank: bs_piwik
- Config: /var/www/stats.technikkultur-erfurt.de/config/config.ini.php

/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;
}
}
```

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;**

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'
);
```

/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;

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

ssl_prefer_server_ciphers on;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_ciphers "ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256:AES256-SHA:AES128-SHA:DES-CBC3-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!MD5:!PSK:!RC4";

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

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_stapling on;
ssl_stapling_verify on;
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;
}
}

```

- → Browser → <https://mail.bytespeicher.org/install>
- `rm -rf /var/www/mail.bytespeicher.org/installer/`

users.bytespeicher.org

/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;
    }
}

```

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:

.duply/mape2k-backup/conf

```
# GPG_KEY='_KEY_ID_'
GPG_PW=''

GPG_KEY_SIGN='58252DC6'
GPG_KEYS_ENC='DD379EDC'
GPG_PW_SIGN='XXXXXXXXXXXXXXXXXX'

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
```

.duply/mkzero-backup/conf

```
# GPG_KEY='_KEY_ID_'
GPG_PW=''

GPG_KEY_SIGN='58252DC6'
GPG_KEYS_ENC='DD379EDC'
GPG_PW_SIGN='XXXXXXXXXXXXXXXXXX'

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
```

Verzeichnisausnahmen:

.duply/mape2k-backup/exclude

```
+ /tmp/mysqldump
- /dev
- /sys
- /proc
- /run
- /tmp
- /var/tmp
- /root/.cache
- /root/backup
```

Benutzer für Sicherung der Datenbank einrichten:

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;
```

Zusätzliche Sicherung der Datenbanken vor der Datensicherung:

.duply/mape2k-backup/pre

```
mkdir -p /tmp/mysqldump
/usr/local/bin/mysql-dump
```

.duply/mape2k-backup/post

```
/bin/rm -rf /tmp/mysqldump
```

/usr/local/bin/mysql-dump.cnf

```
[client]
user=backup
password="PASSWORT"
host=localhost
```

Sicherung per Cronjob:

/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
```

Dauerhafter Link zu diesem Dokument:

<https://wiki.technikkultur-erfurt.de/dienste:bytecluster0001?rev=1491932830>

Dokument zuletzt bearbeitet am: **11.04.2017 17:47**

Verein zur Förderung von Technikkultur in Erfurt e.V

<https://wiki.technikkultur-erfurt.de/>

