

# Auto-delete guest comments

In case you have a guest login enabled and don't want to accept guest spamming you can prevent it using the following bash script with cron trigger (running every 10 minutes). Guest comments are even useless because each guest can delete the guest comments from another guest session. So nobody can guarantee that those will exist a longer time. Deleting such stuff helps to keep clean useful documents which were **not created by guests** but regular users who wanted to put them to **public**.

## Comment deletion

This script is looking within a time fence of 10 minutes. If the script skipped in the meantime, it's possible that comments were overlooked. They have to be cleaned manually then.

```
vim /opt/teedy-clean-comments.sh
```

```
#!/bin/bash
#check for comments which have been created the last 10 minutes. if result is not empty we
send a new email
DB_USER="db_user"
DB_NAME="db_name"
OUT=$(psql -t -U$DB_USER $DB_NAME --no-align --command="
    SELECT
        t_document.doc_title_c,
        t_comment.com_content_c,
        t_comment.com_createdate_d || '\n'
    FROM t_comment
    JOIN t_user ON t_comment.com_iduser_c = t_user.use_id_c
    JOIN t_document ON t_comment.com_iddoc_c = t_document.doc_id_c
    WHERE
        t_document.doc_deletedate_d IS NULL AND
        t_comment.com_deletedate_d IS NULL AND
        t_user.use_username_c = 'guest' AND
        t_comment.com_createdate_d + interval '10 minute' >= now()
    ;
")
```

```

if [[ ! -z $OUT ]]; then
    #echo -e -n _${OUT}_
    #first inform about the comment via mail
    echo -e -n " "$OUT | mail -s "dms.yourdomain.de guest comments" post@fix.de

    OUT=$(psql -t -U$DB_USER $DB_NAME --no-align --command="
        SELECT
            t_comment.com_id_c
        FROM t_comment
        JOIN t_user ON t_comment.com_iduser_c = t_user.use_id_c
        JOIN t_document ON t_comment.com_iddoc_c = t_document.doc_id_C
        WHERE
            t_document.doc_deletedate_d IS NULL AND
            t_comment.com_deletedate_d IS NULL AND
            t_user.use_username_c = 'guest' AND
            t_comment.com_createdate_d + interval '10 minute' >= now()
        ;
    ")

    #echo $OUT

    BASE_URL="https://dms.yourdomain.de"
    BASE_URL="http://localhost:8080/dms"
    TEEDY_USER="teedy"
    AUTH_TOKEN=$(psql -t -U$DB_USER $DB_NAME --command="SELECT aut_id_c FROM
t_authentication_token AS A JOIN t_user AS U ON U.use_id_c = A.aut_iduser_c WHERE
use_username_c = '$TEEDY_USER' AND aut_lastconnectiondate_d IS NOT NULL LIMIT 1;")
    if [ -z "$AUTH_TOKEN" ]
    then
        echo "NO AUTHTOKEN. Please create a session for the user first to automate things!"
    >&2 #print to stderr to trigger cron.d mail on error
        exit 1
    else
        for VAR in $OUT; do
            curl --silent -X DELETE -H "Cookie: auth_token=$AUTH_TOKEN"
"$BASE_URL/api/comment/$VAR" -k
        done
    fi
else
    echo "Nothing to send and nothing to fix ..."

```

fi

cron.d script in `/etc/cron.d/teedy-clean-comments`

```
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
*/10 * * * * root /opt/teedy-clean-comments.sh > /dev/null
```

You can also directly perform a drop statement by SQL instead (but that might hurt the audit log)

```
DELETE FROM t_comment WHERE com_id_c IN (
SELECT
com_id_c
FROM t_comment
JOIN t_user ON t_comment.com_iduser_c = t_user.use_id_c
JOIN t_document ON t_comment.com_iddoc_c = t_document.doc_id_c
WHERE
t_document.doc_deletedate_d IS NULL AND
t_comment.com_deletedate_d IS NULL AND
t_user.use_username_c = 'guest' AND
t_comment.com_createdate_d + interval '1 minute' >= now())
;
```

---

Version #2

Erstellt: 15 Mai 2025 09:58:17 von Mario Voigt

Zuletzt aktualisiert: 15 Mai 2025 10:01:46 von Mario Voigt