

Running Syncs with inotify

The following script can be used to run the TapeTrack Synchronization command whenever a file is uploaded with FTP or SFTP.

The script does the following:

1. Listens using `inotify` for a change to the directory `/var/sftp/user/upload`.
2. Spawns a separate thread to run the synchronization.
3. Writes output to the reports directory with a time and date stamp.
4. Loops back to listen for the next uploaded file.

```
<sxh bash;> #!/bin/bash
```

```
function control_c {
```

```
    logger -s -t TapeTrackSyncWatcher Request to end TapeTrack Sync Watcher
    recieved
    exit

```

```
};
```

```
set -e
```

```
trap control_c SIGINT trap control_c SIGTERM
```

```
logger -s -t TapeTrackSyncWatcher Starting TapeTrack Sync Watcher
```

```
outfile=$(mktemp "${TMPDIR:-/tmp}/syncwatch.XXXXXXXXXXX")
```

```
while true; do
```

```
    #
    # Wait on a change to the sftp directory
    #
    inotifywait -r -e close_write /var/sftp > $outfile 2> /dev/null

```

```
    #
    # A change has been made so read the results into 3 variables (path, type,
    file)
    #
    read -r r1 r2 r3 < $outfile

```

```
    #
    # Delete the temp file
    #
    rm $outfile

```

```
base=`cut -d'/' -f-4 <<< $r1`
```

```
if grep -q "/upload/" <<< $r1 ; then
    logger -s -t TapeTrackSyncWatcher Launching Sync fpr file $r1$r3
    now=$(date +"%m_%d_%Y_%H_%M_%S")
    syncfile=$r1$r3 TMSS10Sync -d $base/etc/default.ttidef >
$base/reports/stdout-$now.txt 2> $base/reports/stderr-$now.txt &
else
    logger -s -t TapeTrackSyncWatcher Skipping file update for file $r1$r3
fi

done

exit </sxh>
```

From:

<https://docs.tapetrack.com/> - **TapeTrack Documentation**

Permanent link:

<https://docs.tapetrack.com/technote/syncwatcher?rev=1555121173>

Last update: **2025/01/21 22:07**

