

Tvheadend - Bug #6318

Wrong executable spawned for pipe input source

2023-12-31 09:27 - M H

Status:	New	Start date:	2023-12-31
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	IPTV	Estimated time:	0.00 hour
Target version:		Affected Versions:	
Found in version:	4.3-2187~gfd8b9e8ba		

Description

In case of an IPTV-URL with pipe and a non-full-qualified-path for ffmpeg-executable it isn't guaranteed that the correct executable is found:

Example:

1. I have the following stream URL:

```
pipe://ffmpeg -hide_banner -loglevel fatal -i https://zdf-hls-18.akamaized.net/hls/live/2016501/dach/veryhigh/master.m3u8 ...
```

2. the executable for ffmpeg is located in /usr/bin

```
$ which ffmpeg
/usr/bin/ffmpeg
```

3. In /usr/bin there are other executables starting with 'ffmpeg...', i.e.

```
$ ls -l ffm*
-rwxr-xr-x 1 root root 301544 Dez 30 12:30 ffmpeg
-rwxr-xr-x 1 root root 22920 Dez 30 12:29 ffmpegthumbnailer
```

4. I have seen in /var/log/syslog that tvheadend spawns '/usr/bin/ffmpegthumbnailer' instead of '/usr/bin/ffmpeg'

```
syslog.1:Dec 30 12:53:54 heckies-nuc tvheadend[16726]: spawn: Executing "/usr/bin/ffmpegthumbnailer"
```

My guess is that the problem is in spawn.c --> find_exec

```
/*
 * Search PATH for executable
 */
int
find_exec ( const char *name, char *out, size_t len )
{
    int ret = 0;
    char bin[512];
    char *path, *tmp, *tmp2 = NULL;
    DIR *dir;
    struct dirent *de;
    struct stat st;
    if (name[0] == '/') {
        if (lstat(name, &st) return 0;
        if (!S_ISREG(st.st_mode) || !(st.st_mode & S_IEXEC)) return 0;
        strncpy(out, name, len);
        return 1;
    }
    if (!(path = getenv("PATH"))) return 0;
    path = strdup(path);
```

```

tmp = strtok_r(path, ":", &tmp2);
while (tmp && !ret) {
    if ((dir = opendir(tmp)) {
        while ((de = readdir(dir)) {
            if (strstr(de->d_name, name) != de->d_name) continue; // <----- comparison wrong?
            snprintf(bin, sizeof(bin), "%s/%s", tmp, de->d_name);
            if (lstat(bin, &st) continue;
            if (!S_ISREG(st.st_mode) || !(st.st_mode & S_IEXEC)) continue;
            strcpy(out, bin, len);
            ret = 1;
            break;
        }
        closedir(dir);
    }
    tmp = strtok_r(NULL, ":", &tmp2);
}
free(path);
return ret;
}

```

Workaround is to use full-qualified path like this:

```

pipe:///usr/bin/ffmpeg -hide_banner -loglevel fatal -i https://zdf-hls-18.akamaized.net/hls/live/2016501/dach/veryhigh/master.m3u8 ...

```