

Contents

Contents.....	1
Parental Control on Linux: Time Restriction.....	1
pam_time.....	1
Warnings and Forced Logout.....	2

Parental Control on Linux: Time Restriction

Updated: 29th March 2016

With Linux becoming more popular on the desktop, it is worth considering how parents can control their children's computer usage. This article describes one method of restricting the times at which particular users can use a computer. The method was developed on Ubuntu, but it should be applicable to many other Linux distributions. It requires modifying the system configuration files, so root access (e.g. using sudo) is necessary. If you do not know how to get root access, these instructions are not for you.

The time control is achieved in three parts:

- Access Time Control: The pam_time module is used to control when particular users can log in.
- Forced Logout: The killall command is used to terminate all programs being run by a particular user.
- Warnings: The notify-send command is used to warn the user that they are about to be forced out.

pam_time

PAM (Pluggable Authentication Modules) are the heart of Linux authentication, see [an introduction to PAM](#) to find out more about it. In short, PAM allows an administrator a great deal of flexibility in configuring how users are authenticated and granted access to a system. To enable time restrictions, it is necessary to create a file containing the restrictions and to tell PAM to apply the restrictions. **Warning:** Errors in PAM configuration files can lock you out of the system. One method for recovering from such a lockout would be to boot from a Linux CD, mount the problem filesystem and edit the configuration.

Edit the file `/etc/security/time.conf` to specify the time restrictions. This file should only be writable by root - you don't want your users changing their own restrictions. The syntax of the file is describe in the man page:

```
man time.conf
```

Example:

```
*;*;harry;A11700-2100
*;*;sally|peter;Wd0000-2400|Wk1800-0800
```

User harry is permitted access every day from 5pm to 9pm and users sally and peter are permitted access anytime at weekends and outside of 8am to 6pm on week days.

Edit the file `/etc/pam.d/common-account`, add this line at the end:

```
account required pam_time.so
```

This enables the access restrictions.

Warnings and Forced Logout

While PAM can be used to enforce access restrictions, once a user has access they can continue using the system. Therefore, another method must be used to force the user off. It is only polite to warn the user that they are about to be forced off, and to allow them time to save their work. The `notify-send` command can be used to pop up a warning, the command can be installed thus:

```
apt-get install notify-osd
```

This script uses `notify-send` to send warnings at 10 minutes, 5 minutes, one minute, 30 seconds and 10 seconds before forcing the user off by using `killall` to kill all the processes they own:

```
#!/usr/bin/perl
#
# warneject.pl
# Warns a user several times they are about to be logged off, then does
  it.

use warnings;
use strict;

use vars qw( $who $killall $notifysend $icon %cusers %kusers
  %excludedusers );

$who= '/usr/bin/who';
$killall= '/usr/bin/killall';
$notifysend= '/usr/bin/notify-send';
$icon= '/usr/share/icons/Humanity/actions/32/stop.svg';

%excludedusers = ( 'root' => 1); # Never kill yourself!

unless (open WHO, "$who |") {
  die "Failed to run who: $!";
}

while () {
  /^(\w+)/;
  next if exists $excludedusers{$1};
  $cusers{$1}++;
}

foreach my $usr (@ARGV) {
  next unless ($usr =~ /^(\w+$/);
  next if ($excludedusers{$usr});
  $kusers{$usr}++ if $cusers{$usr};
}

unless (%kusers) {
  print STDERR "No users to eject.\n";
  exit;
}
print STDERR "Users to eject: ", join(' ',keys %kusers), "\n";
```

```

notify( 'WARNING: Session Expired', "Your login session has expired, you
  will be logged out in 10 minutes. Please save your work and exit your
  applications.");
sleep( 300);
notify( 'WARNING: Session Expired', "Your login session has expired, you
  will be logged out in 5 minutes. Please save your work and exit your
  applications.");
sleep( 240);
notify( 'URGENT WARNING: Session Expired', "Your login session has
  expired, you will be logged out in ONE minute. Please save your work
  and exit your applications.");
sleep( 30);
notify( 'URGENT WARNING: Logout in 30 seconds!', "Make sure your work is
  safe.");
sleep( 20);
notify( 'VERY URGENT: 10 Seconds to Logout!', "Time to go.");
sleep( 10);
notify( "Time's Up", "Logging out.");
ejectuser();

exit;

sub notify {
  my ($ttl, $msg)= @_ ;
  `DISPLAY=:0 XAUTHORITY=/var/run/lightdm/root/:0 $notifysend -u critical
  -i $icon "$ttl" "$msg"`;
}

sub ejectuser {
  foreach my $usr (keys %kusers) {
    `killall -u $usr`;
  }
}

```

If it is saved as /root/bin/warneject.pl and made executable, then it can be run from /etc/crontab at a suitable time. To complete the example above, add these lines to /etc/crontab:

```

0 21 * * * root /root/bin/warneject.pl harry 0 8 * * 1-5 root
  /root/bin/warneject.pl sally peter

```

User harry will be warned at 9pm every day, and users sally and peter will be warned at 8am weekdays. They will be forced off ten minutes later.

More Information

- [How PAM works](#)
- [How do I restrict my kids' computing time?](#)
- [send message to user's GUI through command line](#)
- [Notify-send HOWTO](#)
- [Notify OSD](#)



Suite C & D, 8/F, Yally Industrial Building
 6 Yip Fat Street, Wong Chuk Hang, Hong Kong
 Tel: 2870 8550 Fax: 2870 8563
 E-mail: info@yuik.com.hk
<http://www.yuik.com.hk/>

<http://www.yuikee.com.hk/>

- Security Software Support & Distribution

- Anti-Virus

- Anti-Spam

- Encryption

E-Learning

- Content & Curriculum Development

- Training

Security

*Your
Peace of Mind
Is Our
Commitment*

- Information Security Consultancy

- Alert Services & Web Monitoring

- Ethics, Safety & Security

Education

- Project Development & Management

- Educational Software Distribution

<http://education.yuikee.com.hk/>