Cron

Cron is the name of a Unix program that runs predefined tasks on a computer at regular intervals. It assists some of Totara modules to perform tasks on a scheduled basis. For example, the cron process might tell Totara to check all discussion forums so it can mail out copies of new posts to people who have subscribed to that forum.

The primary Totara script that does all this is located in the admin directory, and is called cron.php. You need to set up a mechanism where this script is run regularly (eg every five or ten minutes). The script can perform functions at periods defined by each module. This kind of regular mechanism is known as a cron service.

Starting Cron

In Totara there are a number of way to invoke cron.php. Cron can be started from the address bar in a browser (URL usually ending in admin/cron.php), via a Daemon, or wq1et, curl or some other code.

The cron process can be protected by a password.

The cronclionly Security > Site Policies setting with stop the browser URL trigger. This sets Totara so that cron.php cannot be invoked by the Totara URL.
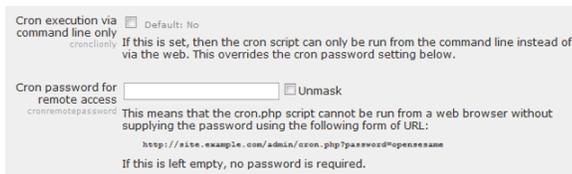
Cron execution via command line only cronclionly	☐ Default: No If this is set, then the cron script can only be run from the command line instead of via the web. This overrides the cron password setting below.
Cron password for remote access cronremotepassword	☐ Unmask This means that the cron.php script cannot be run from a web browser without supplying the password using the following form of URL: http://site.example.com/admin/cron.php?password=opensesame If this is left empty, no password is required.

Cron service location and timing

Note that the machine performing the cron does not need to be the same machine that is running Totara. For example, if you have a limited web hosting service that does not have a cron service, then you might choose to run cron on another server or on your home computer. All that matters is that the cron.php file is called regularly.

The load of this script is not very high, so 5 minutes is usually reasonable, but if you're worried about it you can reduce the time period to something like 15 minutes or even 30 minutes. It's best not to make the time period too long, as delaying mail-outs can slow down activity within the course. Remember that mail-outs also wait for the editing time to expire before being queued for sending.

Testing cron and manual trigger

First, test that the script works by running it directly from your browser: http://example.com/totara/admin/cron.php

If cron is called from the command line by any user logged in to your Totara it will create a temporary admin environment in order to run and then log the user out. You can disable command line running of cron by disabling the appropriate section in the cron.php file.

Now, you need to set up some of way of running the script automatically and regularly.