Daily reports

aily Reports is a new feature in Tiki4, and improved in Tiki8 and Tiki9 that informs you about changes on sites you ar watching with one periodic email.

It is a summarised version of the normal notifications and it replaces them.

Enabling the feature

User Watches and Daily Reports have to be enabled. I recommend to enable the My Tiki menu too. In Tiki6, for instance, you can enable them here:

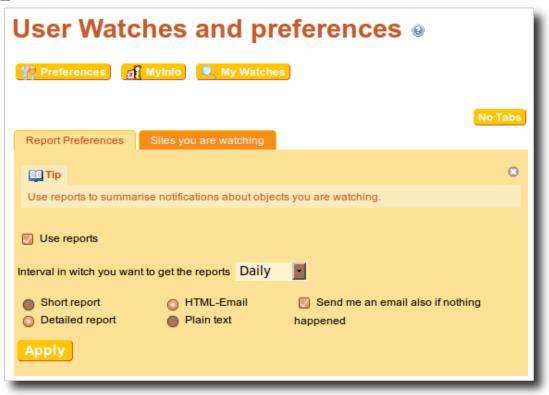
- Admin Home -> Features -> Show My tiki: check
- Admin Home -> Features -> User Watches: check
- Admin Home -> Features -> Daily Reports: check

Since Tiki8, at least, all three features are found in "Admin Home \rightarrow Community", and Show My tiki is only found in "Admin Home \rightarrow Community"

If User Watches and Daily Reports are enabled, each user can decide to enable reports, or to further use normal notifications.

Usage

To use reports, go to My Watches in the main application Menu from your tiki site (**MyTiki > My Watches** - **tiki-user_watches.php**) and adjust your preferences to fit your needs.



Prepare the cron job

The next step is to prepare the cron job.

Before Tiki6:

Open tiki-user_reports_send.php and link \$tikiUrl in line 15 to your Tikiwiki installation
 Since Tiki6 this step is not needed any more.

Since Tiki9:

- You can call the cron job from external servers, by means of using a token to allow the cron job to be launched remotely.
- You can also, as always, launch the cron job locally from the same server where the tiki is hosted, without using any token.

Since Tiki12.3:

- You can use Console (console.php script) to manage the daily reports on the command line. Therefore, you can now use this feature in MultiTiki setups, appending the param like --site=sitename.example.com.

Then you can enable a cronjob. Example for a cronjob that calls the script every day at 3 am:

Since Tiki12.3

00 3 * * * cd /var/www; php console.php daily-report:send

Or

00 3 * * * cd /var/www; php console.php d:s

Replace /var/www with your /path/from/root/to/tiki

In case of Tikis with domains site1.example.com & site2.example.com, in a MultiTiki setup, append an argument like -site=sitename.example.com

00 3 * * * cd /var/www;php console.php d:s --site=site1.example.com 00 3 * * * cd /var/www;php console.php d:s --site=site2.example.com

The above may not work on all server configurations. You are encouraged to test the command first on the console at the server, to check that there is no error message or suspicious warning. If you can't run it this way, the cron may be called through the web server.

00 3 * * * wget -O /tmp/out http://localhost/tiki-user reports send.php

or something like

00 3 * * * /usr/local/bin/wget -O - -q -t 1 http://localhost/tiki-user_reports_send.php

or even with some command based on the other command **curl** and an associated command (like "mutt", which need to be installed separatedly) to send the output of the cron job to an email account. See the exmple below.

Token access for anonymous users

And since Tiki9, you can use a token in the url to call, in order to allow the cron script to be launched from an externa server. An example of using the token and sending the output by email at 5, 11, 17 and 23h each day is shown below

0 5,11,17,23 * * * curl

'http://externaldomain.org/tiki-user_reports_send.php?TOKEN=98762542c722d08b14bb54b029389612' 2> /tmp/output.txt | mutt -s "[FOO]tiki-user_reports_send.php?TOKEN=... error output" foo@bar.org < /tmp/output.txt

For wget to work in virtual servers, you need to ensure that you have the package xterm installed.

How it works internally

lib/reportslib.php

Contains the main class and all important code for reports.

tiki-user reports.php

This file is used to enable or disble reports and to set the report preferences. It catches POST requests from tikiuser_watches.php

tiki-user_reports_send.php

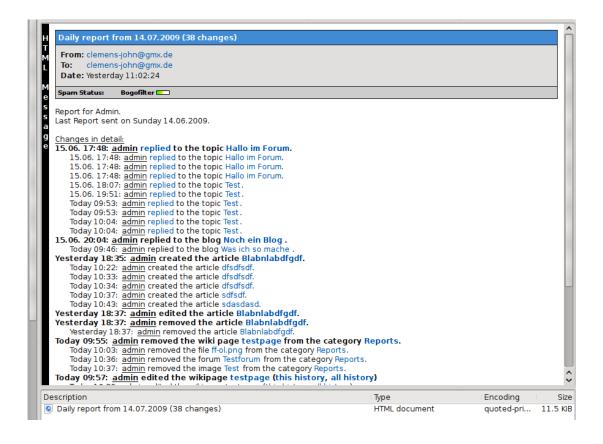
This file is the file that initiates sending reportemails. It can and may only be called by the server.

If you want to use reports before Tiki6, open the file and set \$tikiUrl in line 15 to your Tikiwiki installation.

The last step is to enable a cronjob that calls this file.

Once there are some actions done that would send emails to the user, those action leave a record in the mysql table tiki_user_reports_cache, and when the cron job is run, the records in this table will be used to feed the report to be sent to the user by email.

Example of a Html report:



Changing groups of users to receive daily reports by default

It was not until Tiki6.5, 7.3 and 8.0 that a tiki admin could indicate that new users receive daily digests of notification emails by default.

Therefore, many live tiki sites using previous versions of the software since long ago, with potentially hundreds of users already, might not benefit from that performance improvement untill all users manually indicated that they wanted to receive daily digests instead of instantaneous emails. However, there is some performance limitation in tik when you have hundreds of users receiving instantaneous notification emails each time a user adds some new content. This can mean moving from a couple of seconds, on a shared hosting, to have tiki back ready after sending some new coment or traker item, for instance, into lasting 45 seconds before tiki is ready back after sending all the emails associated to that same action (posting a new comment or tracker item, in this example)

Therefore, it's not rare that an admin would want to change at once a whole group of users from receiving instantaneous emails, to receiving them grouped daily based on a cron job. This change alleviates tiki from sending so many emails with every new action, and brings tiki back to a fast performance from the end user point of view.

This change can be applied directly with a SQL query on the database (through phpmyadmin or by other means). If your groups of users is "foo group", then the SQL query to run directly in your mysql db is:

sql code to run directly on the on mysql database

INSERT INTO `tiki_user_reports` (`user`, `interval`, `view`, `type`, `time_to_send`, `always_email`, `last_report`)

(SELECT uu.`login`, 'daily', 'detailed', 'html', '0000-00-00 00:00:00', 1, NOW() FROM `users_users` uu,

`users_usergroups` uug WHERE uug.`groupName` = 'foo_group' AND uu.`userld` = uug.`userld` AND uu.`login`

NOT IN (SELECT user FROM tiki_user_reports));

Related

User Watches Action log alias names for this page

Daily Reports for User Watches | DailyReports | PeriodicReports | Periodic Reports | Daily digest | DailyReport | Daily Report