

Email Notification of Ticket Changes

Trac supports notification of ticket changes via email.

Email notification is useful to keep users up-to-date on tickets/issues of interest, and also provides a convenient way to post all ticket changes to a dedicated mailing list. For example, this is how the [?Trac-tickets](#) mailing list is set up.

Disabled by default, notification can be activated and configured in [trac.ini](#).

Receiving Notification Mails

When reporting a new ticket or adding a comment, enter a valid email address or your Trac username in the *reporter*, *assigned to/owner* or *cc* field. Trac will automatically send you an email when changes are made to the ticket, depending on how notification is configured.

How to use your username to receive notification mails

To receive notification mails, you can either enter a full email address or your Trac username. To get notified with a simple username or login, you need to specify a valid email address in the *Preferences* page.

Alternatively, a default domain name (`smtp_default_domain`) can be set in the [TracInI](#) file, see [Configuration Options](#) below. In this case, the default domain will be appended to the username, which can be useful for an "Intranet" kind of installation.

When using apache and mod_kerb for authentication against Kerberos / Active Directory, usernames take the form (username@EXAMPLE.LOCAL). To avoid this being interpreted as an email address, add the Kerberos domain to (ignore_domains).

Ticket attachment notifications

Since 1.0.3 Trac will send notifications when a ticket attachment is added or deleted. Usually attachment notifications will be enabled in an environment by default. To disable the attachment notifications for an environment the `TicketAttachmentNotifier` component must be disabled:

```
[components]
trac.ticket.notification.TicketAttachmentNotifier = disabled
```

Configuring SMTP Notification

Important: For [TracNotification](#) to work correctly, the `[trac] base_url` option must be set in [trac.ini](#).

Configuration Options

These are the available options for the [notification] section in trac.ini:

[notification]

[illegible]

[notification-subscriber]

The notifications subscriptions are controlled by plugins. All `INotificationSubscriber` components are in charge. These components may allow to be configured via this section in the `trac.ini` file.

See [TracNotification](#) for more details.

Available subscribers:

Subscriber	Description
<code>AlwaysEmailSubscriber</code>	
<code>CarbonCopySubscriber</code>	Ticket that I'm listed in the CC field is modified
<code>NewTicketSubscriber</code>	Any ticket is created
<code>TicketOwnerSubscriber</code>	Ticket that I own is created or modified
<code>TicketPreviousUpdatersSubscriber</code>	Ticket that I previously updated is modified
<code>TicketReporterSubscriber</code>	Ticket that I reported is modified
<code>TicketUpdaterSubscriber</code>	I update a ticket

Example Configuration (SMTP)

```
[notification]
smtp_enabled = true
smtp_server = mail.example.com
smtp_from = notifier@example.com
smtp_replyto = myproj@projects.example.com
```

```
smtp_always_cc = ticketmaster@example.com, theboss+myproj@example.com
```

Example Configuration (sendmail)

```
[notification]
smtp_enabled = true
email_sender = SendmailEmailSender
sendmail_path = /usr/sbin/sendmail
smtp_from = notifier@example.com
smtp_replyto = myproj@projects.example.com
smtp_always_cc = ticketmaster@example.com, theboss+myproj@example.com
```

Subscriber Configuration

The default subscriptions are configured in the [notification-subscriber] section in trac.ini:

[notification-subscriber]

The notifications subscriptions are controlled by plugins. All INotificationSubscriber components are in charge. These components may allow to be configured via this section in the trac.ini file.

See [TracNotification](#) for more details.

Available subscribers:

Subscriber	Description
AlwaysEmailSubscriber	
CarbonCopySubscriber	Ticket that I'm listed in the CC field is modified
NewTicketSubscriber	Any ticket is created
TicketOwnerSubscriber	Ticket that I own is created or modified
TicketPreviousUpdatersSubscriber	Ticket that I previously updated is modified
TicketReporterSubscriber	Ticket that I reported is modified
TicketUpdaterSubscriber	I update a ticket

Each user can override these defaults in his *Notifications* preferences.

For example to unsubscribe from notifications for one's own changes and comments, the rule "Never notify: I update a ticket" should be added above other subscription rules.

Customizing the e-mail subject

The e-mail subject can be customized with the `ticket_subject_template` option, which contains a [?Genshi text template](#) snippet. The default value is:

```
$prefix #${ticket.id}: $summary
```

The following variables are available in the template:

- `env`: The project environment (see [?env.py](#)).
- `prefix`: The prefix defined in `smtp_subject_prefix`.
- `summary`: The ticket summary, with the old value if the summary was edited.
- `ticket`: The ticket model object (see [?model.py](#)). Individual ticket fields can be addressed by appending the field name separated by a dot, eg `$ticket.milestone`.

Customizing the e-mail content

The notification e-mail content is generated based on `ticket_notify_email.txt` in `trac/ticket/templates`. You can add your own version of this template by adding a `ticket_notify_email.txt` to the templates directory of your environment. The default looks like this:

```
$ticket_body_hdr
$ticket_props
{% choose ticket.new %}\
{% when True %}\
$ticket.description
{% end %}\
{% otherwise %}\
{% if changes_body %}\
${_('Changes (by %(author)s):', author=change.author)}

$changes_body
{% end %}\
{% if changes_descr %}\
{% if not changes_body and not change.comment and change.author %}\
${_('Description changed by %(author)s:', author=change.author)}
{% end %}\
$changes_descr
--
{% end %}\
{% if change.comment %}\

${changes_body and _('Comment:') or _('Comment (by %(author)s):', author=change.author)}

$change.comment
{% end %}\
{% end %}\
{% end %}\

--
${_('Ticket URL: <(link)s>', link=ticket.link)}
$project.name <${project.url or abs_href()}>
$project.descr
```

Sample Email

#42: testing

```

-----+-----
      Id: 42          |      Status: assigned
Component: report system |      Modified: Fri Apr 9 00:04:31 2004
Severity: major        |      Milestone: 0.9
Priority: lowest        |      Version: 0.6
Owner: anonymous        |      Reporter: jonas@example.com
-----+-----

```

Changes:

```

* component: changeset view => search system
* priority: low => highest
* owner: jonas => anonymous
* cc: daniel@example.com =>
    daniel@example.com, jonas@example.com
* status: new => assigned

```

Comment:

I'm interested too!

--

Ticket URL: <http://example.com/trac/ticket/42>

My Project <http://myproj.example.com/>

Customizing e-mail content for MS Outlook

MS Outlook normally presents plain text e-mails with a variable-width font, and as a result the ticket properties table will most certainly look like a mess in MS Outlook. This can be fixed with some customization of the [e-mail template](#).

Replace the following second row in the template:

\$ticket_props

with this (requires Python 2.6 or later):

```

-----
{% with
    pv = [(a[0].strip(), a[1].strip()) for a in [b.split(':') for b in
        [c.strip() for c in
            ticket_props.replace('|', '\n').splitlines()[1:-1]] if ':' in b]];
    sel = ['Reporter', 'Owner', 'Type', 'Status', 'Priority', 'Milestone',
        'Component', 'Severity', 'Resolution', 'Keywords'] %}\
${'\n'.join('%s\t%s' % (format(p[0]+' ', '<12'), p[1]) for p in pv if p[0] in sel)}
{% end %}\
-----

```

The table of ticket properties is replaced with a list of a selection of the properties. A tab character separates the name and value in such a way that most people should find this more pleasing than the default table when using MS Outlook.

#42: testing

```

Reporter: jonas@example.com
Owner: anonymous
Type: defect
Status: assigned
Priority: lowest
Milestone: 0.9
Component: report system
Severity: major
Resolution:
Keywords:

```

Changes:

```
* component: changeset view => search system
* priority: low => highest
* owner: jonas => anonymous
* cc: daniel@example.com =>
    daniel@example.com, jonas@example.com
* status: new => assigned
```

Comment:

I'm interested too!

--

Ticket URL: <<http://example.com/trac/ticket/42>>

My Project <<http://myproj.example.com/>>

Important: Only those ticket fields that are listed in `sel` are part of the HTML mail. If you have defined custom ticket fields which are to be part of the mail, then they have to be added to `sel`. Example:

```
sel = ['Reporter', ..., 'Keywords', 'Custom1', 'Custom2']
```

However, the solution is still a workaround to an automatically HTML-formatted e-mail.

Using GMail as the SMTP relay host

Use the following configuration snippet:

```
[notification]
smtp_enabled = true
use_tls = true
mime_encoding = base64
smtp_server = smtp.gmail.com
smtp_port = 587
smtp_user = user
smtp_password = password
```

where *user* and *password* match an existing GMail account, ie the ones you use to log in on <http://gmail.com>.

Alternatively, you can use `smtp_port = 25`.

You should not use `smtp_port = 465`. Doing so may deadlock your ticket submission. Port 465 is reserved for the SMTPS protocol, which is not supported by Trac. See [?#7107](#) for details.

Troubleshooting

If you cannot get the notification working, first make sure the log is activated and have a look at the log to find if an error message has been logged. See [TracLogging](#) for help about the log feature.

Notification errors are not reported through the web interface, so the user who submits a change or a new ticket never gets notified about a notification failure. The Trac administrator needs to look at the log to find the error trace.

Permission denied error

Typical error message:

```
...
File ".../smtplib.py", line 303, in connect
    raise socket.error, msg
error: (13, 'Permission denied')
```

This error usually comes from a security settings on the server: many Linux distributions do not allow the web server (Apache, ...) to post email messages to the local SMTP server.

Many users get confused when their manual attempts to contact the SMTP server succeed:

```
telnet localhost 25
```

This is because a regular user may connect to the SMTP server, but the web server cannot:

```
sudo -u www-data telnet localhost 25
```

In such a case, you need to configure your server so that the web server is authorized to post to the SMTP server. The actual settings depend on your Linux distribution and current security policy. You may find help in the Trac [?MailingList](#) archive.

Relevant ML threads:

- SELinux: [?http://article.gmane.org/gmane.comp.version-control.subversion.trac.general/7518](http://article.gmane.org/gmane.comp.version-control.subversion.trac.general/7518)

For SELinux in Fedora 10:

```
$ setsebool -P httpd_can_sendmail 1
```

Suspected spam error

Some SMTP servers may reject the notification email sent by Trac.

The default Trac configuration uses Base64 encoding to send emails to the recipients. The whole body of the email is encoded, which sometimes trigger *false positive* spam detection on sensitive email servers. In such an event, change the default encoding to "quoted-printable" using the `mime_encoding` option.

Quoted printable encoding works better with languages that use one of the Latin charsets. For Asian charsets, stick with the Base64 encoding.

See also: [TracTickets](#), [TracIni](#), [TracGuide](#), [?TracDev/NotificationApi](#)