

Internet.nl Dashboard Documentation

version

internet.nl

March 06, 2023

Contents

internet.nl dashboard documentation	1
Dashboard User Management	1
Overview	1
Abstract	1
In practice	2
Management	2
Account Management	2
User Management	3
Email Templates	5
General information	5
Where can i find templates	5
What templates are available	5
How do i define templates in multiple languages	5
What template system is used	5
What do i need to watch out for	5
What are the pros and cons of the current approach	5
How to set up e-mail sending	6
Scan Finished mail	6
When is this mail sent	6
How to test this mail	6
scan_finished tags	6
Example template	8
Indices and tables	11

internet.nl dashboard documentation

Dashboard User Management

Overview

The dashboard knows of three places that have to do with users:

- The API User
- The Account (which is a name for multiple users)
- The User

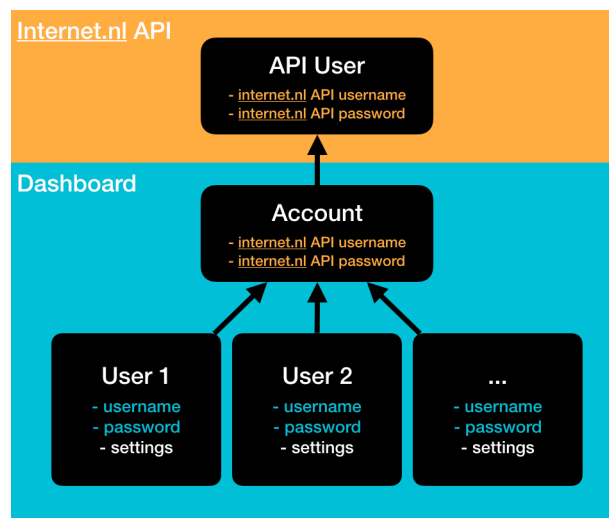
These connect to each other as shown in the following image.

Summarized:

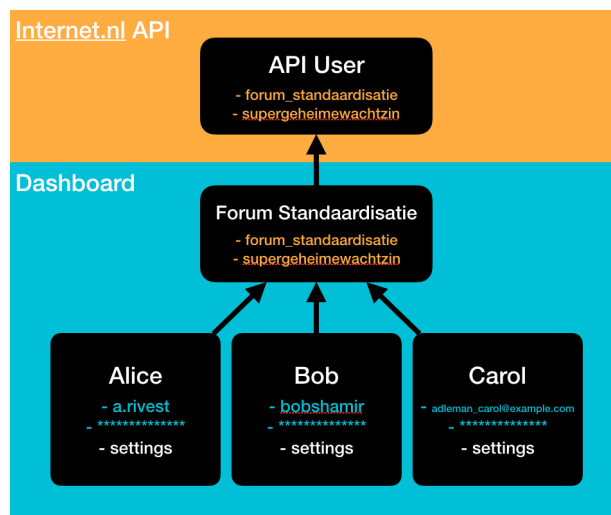
- An account can connect to the Internet.nl API using credentials.
- An account holds domains, report settings, scans and reports
- A User is part of an Account, there can be multiple users for an account
- A User has their own username, password and settings

We know that 'account' and 'user' is ambiguous.

Abstract



In practice



Management

User and account management in the dashboard is done via the Dashboard app and the Django Admin Interface. The latter is a bare bones data management application that allows editing data directly onto the database. The user experience is good enough for system administrators that need ultimate power and flexibility. With great power comes great responsibility.

Account Management

Accounts can be created using the accounts menu. Accounts contain very little information. You will see report settings, which don't have to be altered. Just fill out the name and the Internet.nl API credentials.

The screenshot shows the 'DASHBOARD ADMIN' interface. On the left is a sidebar with navigation links: HOME, VIEW SITE, DOCUMENTATION, USER, Users, Accounts, and TOTP devices. The main content area is titled 'HOME > INTERNET_NL_DASHBOARD > ACCOUNTS'. It features a search bar with the text 'internet' and a 'SEARCH' button, showing '1 result'. Below the search bar are 'Import', 'Export', and '+ Add account' buttons. A table lists accounts with columns: NAME, INTERNET NL API USERNAME, CAN CONNECT TO INTERNET NL API, and NO OF USERS. The table contains one entry: 'Internet.nl' with username 'example api username' and a green checkmark in the 'CAN CONNECT' column. Below the table is a form with a dropdown menu and a 'GO' button. At the bottom right, it says '1 account'.

Here is an example of a filled in account. Notice how passwords here are not hashed, but encrypted. That's because they need to be sent to the API in a way that the API can read it. These credentials are sent over TLS to the API.

HOME > INTERNET_NL_DASHBOARD > ACCOUNTS > ADD ACCOUNT

Name:

Report settings:

This stores reporting preferences: what fields are shown in the UI and so on (if any other). This field can be edited on the report page.

Internet.nl api username:
Internet.nl API Username

New password:
Changing this value will set a new password for this account.

User Management

User management is a bit more challenging. An account can have multiple users. That's what you see in the screenshot below: both users are in the 'Internet.nl' account.

HOME > AUTHENTICATION AND AUTHORIZATION > USERS

2 results

<input type="checkbox"/>	USERNAME x	IN ACCOUNT	FIRST NAME	LAST NAME	EMAIL ADDRESS	ACTIVE	STAFF STATUS
<input type="checkbox"/>	user_1	Internet.nl	First Name	Last Name		✓	✗
<input type="checkbox"/>	user_2	Internet.nl	First	Last	email@example.com	✓	✓

2 users

Editing a user reveals a long form. This form allows editing all the standard user information as well as user information specific to the dashboard experience.

In the top part of the form you can setup a username, password, first name, last name and e-mail.

Permissions only require the 'active' field. If you want to make somebody staff or superuser, they first have to set up their Second Factor authentication after logging in. No Groups of "User Permissions" need to be set up.

The last part of the form are the settings the user can edit when logged into the dashboard. Here you can specify under which account the user operates. You can leave the other fields empty, they can set that up themselves.

HOME > AUTHENTICATION AND AUTHORIZATION > USERS > USER_1

History

Username:*

user_1

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password:

algorithm: pbkdf2_sha256 iterations: 216000 salt: ?k***** hash: tLc *****

Raw passwords are not stored, so there is no way to see this user's password, but you can change the password using [this form](#).

PERSONAL INFO

First name:

First Name

Last name:

Last Name

Email address:

PERMISSIONS

☒ Active

Designates whether this user should be treated as active. Unselect this instead of deleting accounts.

☐ Staff status

Designates whether the user can log into this admin site.

☐ Superuser status

Designates that this user has all permissions without explicitly assigning them.

Groups:

+

The groups this user belongs to. A user will get all permissions granted to each of their groups. Hold down "Control", or "Command" on a Mac, to select more than one.

User permissions:

actstream | action | Can add action

actstream | action | Can change action

Specific permissions for this user. Hold down "Control", or "Command" on a Mac, to select more than one.

IMPORTANT DATES

Last login:

2021-09-07

11:13:53

Date joined:*

2020-01-30

18:44:47

+ Add another Dashboard...

Internet.nl/user_1

DASHBOARD USER: INTERNET.NL/USER_1

Account:*

Internet.nl

Mail preferred mail address:

reports@example.com

This address can deviate from the account mail address for other account features.

Mail preferred language:*

Netherlands

☒ Mail send mail after scan finished

After a scan is finished, an e-mail is sent informing the user that a report is ready.

Mail after mail unsubscribe code:

This is autofilled when sending an e-mail. The user can use mail_send_mail_after_scan_finished to false without logging out.

Notes:

4

Email Templates

The dashboard can send out mail when scans are finished. An email is composed using a template. These templates are rich in functionality, and can be adjusted without the need of re-building and re-deploying this software.

General information

Where can i find templates

Templates are located at `/admin/django_mail_admin/emailtemplate/` on your installation. They are part of the admin system and require the right permissions to be edited. An admin can edit them easily.

What templates are available

Currently there is one feature that uses an e-mail template, which is *scan_finished*. When a scan is finished, this mail template is used to draft a mail.

The scan finished template uses two sub-templates, which are:

- scan_finished
- - detailed_comparison_regression
 - detailed_comparison_improvement

The sub templates are tables that list regressions and improvements.

How do i define templates in multiple languages

Templates are postfixed with a language identifier. The default is `'_en'`, for English. The dashboard currently supports two languages, Dutch and English: `nl` and `en`.

If a template for `_nl` (or any other language) is missing, it will use `_en`.

(Technically there are many solutions to make sure translations are multilingual, most of which require re-compilation or maintenance of external translation files that require a build. When there are many languages in use, there will be a need to switch.)

What template system is used

The Django template system is used, with one exception: all fields from the dashboard are strings.

The Django template system is a mix of special tags and HTML, Javascript and CSS. It is also used in building front end systems which are rendered on the server side.

The Django Template system is documented here: <https://docs.djangoproject.com/en/3.1/ref/templates/language/>

What do i need to watch out for

Syntax mistakes can prevent a mail from being composed or sent. In this case the software will crash. This is the largest drawback of the current approach. When editing a template, be careful.

What are the pros and cons of the current approach

Pros:

- No need to recompile / rebuild / deliver the software when templates change
- Multi language in a simple and straightforward way (allowing changes per language)
- Uses well documented systems of django, html, javascript and css.

Cons:

- No syntax highlighting or checking of templates in the editor, use an external editor(!)

- A template per language is annoying when doing maintenance

How to set up e-mail sending

Email sending uses Django Mail Admin: https://github.com/Bearle/django_mail_admin/

You need to configure an outbox with proper values. You can test if this works via the command line with the command:

```
dashboard send_testmail
```

Scan Finished mail

When is this mail sent

This mail is only sent when the user has set that in their profile. You can review that for example here: </admin/auth/user/1/change/>

The fields that are used to send this mail are:

- Mail preferred mail address: an e-mail address
- Mail preferred language: a language from the list (only supporting NL or EN)
- Mail send mail after scan finished: a boolean value if the mail can be sent
- Mail after mail unsubscribe code: automatically generated

If all preconditions are correct, a mail will be queued as part of the scanning process, just before the scan is finished.

How to test this mail

A test mail can be sent at will on the admin page of AccountInternetNLScan, here: /admin/internet_nl_dashboard/accountinternetnlsan/

A mail can only be sent if a scan is finished and the mail preconditions and setup are correct.

When this mail has been queued, either wait for the periodic task that sends mail is performed (once a minute), or perform this via the command line (during development):

```
dashboard send_queued_mail --processes=1 --log-level=2
```

scan_finished tags

- {{unsubscribe_code}}

Allows an unsubscribe from a specific feed of mails. This code can be used without a login. The url this code is used for is for example:

```
{{dashboard_address}}/spa/#/unsubscribe?feed=scan_finished&unsubscribe_code={{unsubscribe_code}}
```

- {{recipient}}

The recipient of the email, which is, in order of fallback A) the first name, B) the last name, C) the username.

- {{user_id}}

The id of a user, which might be useful at some point.

- {{list_name}}

The name of the domain list that is being scanned.

- {{report_id}}

The mail is about a report. Using this number a link can be built to the report. For example:

```
{{dashboard_address}}/spa/#/report/{{report_id}}
```

- `{{report_average_internet_nl_score}}`

The average score in the report.

- `{{report_number_of_urls}}`

The total number of urls in this report.

- `{{scan_id}}`

The number of the scan performed, which might be useful for context and tracking purposes.

- `{{scan_started_on}}`

The date and time when the scan started, in ISO format.

- `{{scan_finished_on}}`

An approximation of when the scan is finished. This mail is sent as part of the scanning process, which is thus not yet finished. It might be off by a minute or two. This is also in ISO format.

- `{{scan_duration}}`

Number of seconds it took to complete a scan. Also an approximation.

- `{{scan_type}}`

Either web or mail. Can be used in sentences like:

The `{{scan_type}}` scan on `{{list_name}}` is finished.

- `{{previous_report_available}}`

If there is a previous report for this list. The value will be “True” if that is the case. Otherwise it will be “False”. Note that this is a string value, not a boolean value.

- `{{previous_report_average_internet_nl_score}}`

The average score of the previous report. This is used for easy overall comparison.

- `{{compared_report_id}}`

The id of the previous report, can be used to build a link with a comparison, such as:

`{{dashboard_address}}/spa/#/report/{{report_id}}/{{compared_report_id}}`

- `{{comparison_is_empty}}`

A string boolean containing either “True” or “False”. The comparison is empty when all values compared to the previous and current report are the same. There has been no change, at all.

If the comparison is empty, there is no need to show any details of course.

- `{{improvement}}`

The number of improvements made in the current report, compared to the last report.

- `{{regression}}`

The number of regressions in the current report, compared to the last report.

- `{{neutral}}`

The number of neutral values in the current report, compared to the last report. Neutral is either unchanged, or a comparison against an error, not-testable or other hard to compare value.

- `{{comparison_report_available}}`

A simple value to check if a comparison is available. Can be used to enable or disable sections of the email.

- `{{comparison_report_contains_improvement}}`

Set to “True” if there are improvements in the comparison. There might be only improvements and no regressions and vice versa.

- `{{comparison_report_contains_regression}}`

Set to "True" if there are regressions available.

- `{{days_between_current_and_previous_report}}`

The number of days between the current and previous report.

- `{{comparison_table_improvement}}`

This is a rendered section of html, based on the `detailed_comparison_improvement(_en)` template. To use pre-rendered html, use the following in your e-mail, using the word "safe":

`{{comparison_table_improvement|safe}}`

- `{{comparison_table_regression}}`

See `comparison_table_improvement`.

- `{{domains_exclusive_in_current_report}}`

A comma separated string of domains that are available in the current report, but not in the previous report. These are new domains that have been added to the list, usually. There are also edge cases where the domain could not be scanned last time, but it could this time.

- `{{domains_exclusive_in_other_report}}`

A comma separated string of domains that are only available in the previous report. Probably those domains have been deleted from the list of domains during the new scan.

- `{{dashboard_address}}`

The web address of the dashboard. This is configured in the settings at: `/admin/constance/config/`

Example template

`scan_finished_en` takes into account a multitude of situations where there are no scan results.

This template will probably be quickly outdated, but shows how to build a nice template with the fields above.

```
Hi {{recipient}},<br>
<br>
The {{scan_type}} scan on '{{list_name}}' has finished and your report is ready. The average
<br>
View the report at this link: <br>
<a href="{{dashboard_address}}/spa/#/report/{{report_id}}">
    {{dashboard_address}}/spa/#/report/{{report_id}}</a><br>

{% if previous_report_available == "False" %}
<br>
This is the first report for '{{list_name}}'. The next time this list is scanned, a comparis
{% endif %}

{% if previous_report_available == "True" and comparison_is_empty == "True" %}
<br>
A previous report, #{{compared_report_id}}, is available but contains no changes compared to
{% endif %}

{% if previous_report_available == "True" and comparison_is_empty == "False" %}
<br>
<h3>Changes compared to previous report</h3>
Below a summary is given compared to the previous report, #{{compared_report_id}}. The previ
<br>
You can view the comparison in detail on the dashboard at <a href="{{dashboard_address}}/spa
<br>
<h4>Summary of changes:</h4>
<table>
    <tr style='font-weight: bold; text-align: center;'>
```

Email Templates

```
        <td>{{improvement}}</td><td>{{regression}}</td><td>{{neutral}}</td>
    </tr>
    <tr>
        <td>Improvements</td><td>Regressions</td><td>Neutral</td>
    </tr>
</table>
{% endif %}
```

```
{% if previous_report_available == "True" and comparison_report_contains_improvement != "True" %}
<br>
Only neutral changes have been observed, therefore no detailed overview of changes is included.
{% endif %}
```

```
{% if comparison_report_contains_improvement == "True" or comparison_report_contains_regression == "True" %}
{% if comparison_report_contains_improvement == "True" %}
<br>
<h4>Overview of improvements:</h4>
<table style="">
    <tr>
        <th>Domain</th>
        <th>Score</th>
        <th>Improvement(s)</th>
        <th>Metrics improved</th>
    </tr>
    {{comparison_table_improvement|safe}}
</table>
{% endif %}
```

```
{% if comparison_report_contains_regression == "True" %}
<br>
<h4>Overview of regressions:</h4>
<table>
    <tr>
        <th>Domain</th>
        <th>Score</th>
        <th>Regeression(s)</th>
        <th>Metrics regressed</th>
    </tr>
    {{comparison_table_regression|safe}}
</table>
{% endif %}
```

```
{% endif %}
```

```
{% if domains_exclusive_in_current_report != "" %}
<br>
This report includes new domains, which are not included because they could not be compared:
{% endif %}
```

```
{% if domains_exclusive_in_other_report != "" %}
<br>
The following domains have disappeared in the new report, and are thus not included above:
{% endif %}
```

```
<br>
Regards,<br>
The internet.nl dashboard<br>
<br>
[
```

Email Templates

```
<a href="{{dashboard_address}}/spa/#/unsubscribe?feed=scan_finished&unsubscribe_code={{unsubscribe_code}}">unsubscribe</a>
-
<a href="{{dashboard_address}}/spa/#/account">preferences</a>
]

<style>
table th, table td{
    padding: 5px;
}
</style>
```

detailed_comparison_regression_en:

```
{% for record in data %}
<tr style='background-color: {% cycle 'rgba(0,0,0,.05)' 'inherit' %};'>
    <td style="vertical-align: baseline">
        {{ record.url }}
    </td>
    <td style="vertical-align: baseline">
        <a href="{{ record.new.report }}" target="_blank">
            
        </a>
    </td>
    <td style="vertical-align: baseline">
        {{ record.changes.regression }}
    </td>
    <td>
        <ul>
            {% for metric in record.changes.regressed_metrics %}
                <li>{{ metric }}</li>
            {% endfor %}
        </ul>
    </td>
</tr>
{% endfor %}
```

detailed_comparison_improvement_en:

```
{% for record in data %}
<tr style='background-color: {% cycle 'rgba(0,0,0,.05)' 'inherit' %};'>
    <td style="vertical-align: baseline">{{ record.url }}</td>
    <td style="vertical-align: baseline">
        <a href="{{ record.new.report }}" target="_blank">
            
        </a>
    </td>
    <td style="vertical-align: baseline">
        {{ record.changes.improvement }}
    </td>
    <td>
        <ul>
            {% for metric in record.changes.improved_metrics %}
                <li>{{ metric }}</li>
            {% endfor %}
        </ul>
    </td>
</tr>
{% endfor %}
```

Indices and tables

- `genindex`
- `modindex`
- `search`