

## pfSense - Todo #5480

### inconsistent display of default values in fields

11/18/2015 05:20 PM - Jared Dillard

|  |               |                        |            |
|--|---------------|------------------------|------------|
| <b>Status:</b>   | New           | <b>Start date:</b>     | 11/18/2015 |
| <b>Priority:</b>   | Low           | <b>Due date:</b>       |            |
| <b>Assignee:</b>   | Jared Dillard | <b>% Done:</b>         | 0%         |
| <b>Category:</b>   | Web Interface | <b>Estimated time:</b> | 0.00 hour  |
| <b>Target version:</b>   |               | <b>Release Notes:</b>  |            |
| <b>Plus Target Version:</b>  |               |                        |            |
| <b>Description</b>   |               |                        |            |
| Feedback from a user:<br><br>One thing I noticed is that the UI is inconsistent with the display of default values in fields. It's not a new issue, but since one of the goals of 2.3 is to clean up the UI, it's a good time to fix it.<br>In some cases the default is already in the field as if the user put it there (Max Processes).<br>In other cases the default is shown grayed out if the field hasn't been modified by the user (SSH port).<br>Finally, sometimes it's just empty even if there's a default value, and the default is discussed in the description.<br>My preference would be to show the default grayed out if the field hasn't been modified by the user, like it is with the SSH port.<br>But anything would be acceptable as long as it's consistent. |               |                        |            |

#### History

##### #1 - 11/19/2015 01:47 AM - Chris Buechler

I think that inconsistency is all attributable to whether or not that value exists in the default config.xml. Max processes is 2 in the default config for instance, so the page fills in that field from the config as if it were user-configured. SSH port doesn't exist in config.xml unless the user defines it, so it's blank unless user-configured.

Some would argue none of the default values should be in the config, so it's as easy as possible to see non-defaults that are configured and the config's relatively short (like Cisco IOS). Others would argue all of the defaults should be in the config so all the configuration options are visible.

There should be consistency in that regard, but that's a pretty significant undertaking beyond anything to do with Bootstrap.

##### #2 - 11/19/2015 09:15 AM - Anonymous

Some programmers have used the "placeholder" attribute to put a suggested value into blank fields. e.g.: MAC xx:xx:xx:xx:xx:xx

Others (me included) find the help text a better way to do this. We could consider disabling the placeholder mechanism globally if people find it annoying/confusing.

##### #3 - 11/20/2015 07:45 AM - Renato Botelho

- Project changed from Bootstrap to pfSense

- Category set to Web Interface

- Target version set to 2.3

##### #4 - 12/01/2015 03:31 AM - NOYB NOYB

Think my position would be that the config file is not a user interface and therefore it is irrelevant whether or not default values are there.

The UI is where the values are intended to be viewed. I like the fields with default value grayed.

Maybe there could be a config file that contains only default values and if there is no configured value then the default value is read in and displayed as UI placeholder and used for running operation.

Or maybe a special attribute or something could be designated in the config file to identify default values.

##### #5 - 12/02/2015 04:01 AM - Phillip Davis

There are places that will save all the default values into the config when (if) the user first saves the page. I think this is true for a lot of "Advanced settings" pages.

e.g. For Unbound Advanced Settings, if Save has never been pressed on that page (which often happens - the user has no reason to mess with anything there) then those settings are not in the config at all. The back-end code has hard-coded defaults that it uses if the setting(s) is not specified

in the config.

After the user presses Save on Advanced Settings then all the settings are written into the config, even if the user changed only 1 (or none) from the default values.

In the new GUI it should be easy enough to have a flow like:

- a) For fields that have no value currently set, display the default value in grey in the field, waiting for potential user input. For drop-down lists, what to do? Select the default value, but remember that it is just the default and not actually selected by the user?
- b) When saving, just save to the config any fields for which the user has actually entered a value. For drop-down lists, I guess only save the value if it is different to the default.

One the benefits I see of not storing the default value in the config is that it becomes easy to decide if the user has specified a particular value or not. If there are upgrades where we want to change the default, then it becomes easy and there is nothing to upgrade in the config. The back-end code will simply use the new default value because no value is specified in the config.

At the moment, with Unbound Advanced settings, it is not possible to know if the user has specially chosen the value of each setting (and for many of them has specially chosen what happens to also be the default) or if they only wanted 1 special value and "got the lot" because they were on the same page. That makes it harder when doing future upgrades that change a default.

All the default values for these sort of settings can easily be stored in some "inc" file(s) where the code that needs them can get them efficiently. That is another thing that could be done to consolidate some places where the literal default value is in back-end code, GUI default setting and written in some help text. All those should get the default from the same place, to avoid the problem of things getting out-of-whack.

#### **#6 - 03/01/2016 04:59 PM - Chris Buechler**

- *Target version deleted (2.3)*

nothing needs to be done here for 2.3. Not sure anything needs to be done here at any point, though it wouldn't hurt to have consistency with defaults.