Foreman - Feature #21678

Mark required fields conditionally based on other fields

11/15/2017 02:55 PM - Marek Hulán

Status:	New		
Priority:	Normal		
Assignee:			
Category:	Web Interface		
Target version:			
Difficulty:		**Fixed in Releases:**	
Triaged:		**Found in Releases:**	
Bugzilla link:	1324967	**Red Hat JIRA:**	
Pull request:			

Description

"Puppet Environment" is a required field for provisioning new hosts. However, the form does not flag this field as being required. Other required fields have an "*" next to them to denote them as being required.

As it turned out, the puppet environment is required field only when a puppet master is selected on the host.

It would be nice to have a possibility to switch required flag based on the rest of the form. The similar behavior can be found at subnet form and I support other places too. Since we don't want to extend already complicate JS code for host form, I suppose this will be possible after we refactor that or reimplement our form helpers.

Related issues:

Related to Foreman - Bug #7450: We mark not-required fields as required	**Closed**	09/16/2014
Related to Foreman - Feature #20839: implement bookmark creation using react	**Closed**	09/04/2017

History

#1 - 11/15/2017 02:59 PM - Marek Hulán

- Related to Bug #7450: We mark not-required fields as required added

#2 - 11/15/2017 03:00 PM - Marek Hulán

- Related to Feature #20839: implement bookmark creation using react added

#3 - 11/15/2017 09:28 PM - Ohad Levy

- Subject changed from Mark required fields conditionally based on other fields
to Mark required fields conditionally based on other fields

What is the right behavior? Should we not show/disable the field? (E.g don't show environment if puppet master is not selected?) Or rather explicitly say that you want puppet first (and then select both)?

#4 - 01/19/2018 11:10 AM - Marek Hulán

Today we update the host form based on what user selects (e.g. provisioning method). For now I think just hiding the fields until user selected puppet master is good enough. Although that would mean extending existing host form JS. If we go this way, we should look at other more complicated forms, such as interface or subnet forms.