When creating a HTTP proxy the username and password is used verbatim, leading to issues if they contain special characters

Steps to Reproduce:
1. Configure a squid proxy with the following details.
   hostname: squid.example.com
   port: 3128
   proxy_username: proxyuser
   proxy: password: test\satellite@password+
2. Go to Satellite GUI --> Infrastructure --> HTTP Proxies --> Fill up all the details including the proxy_password, in Test URL mention "https://aws.amazon.com" and then click on "Test Connection".
3. With all options filled up, submit the settings.
4. Now open the newly created proxy and look at the password field.

Actual results:
At Step 2, with test connection, you will get an error pop up saying "bad password component"
At Step 4, we will be able to see that the password is blank

Expected results:
Satellite 6.7 should be able to accept the password with special characters properly and allow it to save\user further from GUI.

There are two issues at play here.
The first is on our side. We try to stuff the user provided strings into an URI object, but that fails because special characters are not allowed there. If we escape the user provided strings first (turn '@' into '%40' and so on), this part should be resolved.

The other issue is restclient (the library we use for making http requests) does not unescape the proxy username and password, so even if we fix the first issue, then the escaped form will be sent to the proxy and the proxy will reject it. There is a PR opened against restclient which fixes this, but last movement there was in October 2018, so I'm afraid we can't wait for this to be fixed there. Unless I'm missing something the only real option we have is to monkey-patch restclient the same way ManageIQ did.


Associated revisions
Revision 150d024e - 06/11/2021 08:47 AM - Adam Ruzicka
Fixes #32192 - Handle special characters in HTTP proxy credentials

Originally we tried to using user provided credentials verbatim, leading to errors when they contained special characters. This is resolved by escaping them first as required by section 2.4.3 of RFC2396.
Revision a196afb1 - 06/11/2021 08:47 AM - Adam Ruzicka
Fixes #32192 - Monkey patch restclient to unescape proxy credentials

Restclient as of version 2.1.0 uses provided credentials stored in a proxy URI object verbatim. If the credentials are escaped as they should be, then this leads to errors when we try to use the credentials because restclient sends the escaped form.

There is a PR in upstream restclient, which addresses this issue, but last movement there was in October 2018. ManageIQ folks decided to monkey patch restclient when they ran into the same issue, this commit is a loose adaptation of the original and ManageIQ patches.


Revision a41cabb4 - 06/11/2021 08:47 AM - Adam Ruzicka
Fixes #32192 - Override only the relevant parts

History

#1 - 03/24/2021 12:48 PM - The Foreman Bot
- Status changed from Assigned to Ready For Testing
- Pull request https://github.com/theforeman/foreman/pull/8412 added

#2 - 06/11/2021 08:48 AM - The Foreman Bot
- Fixed in Releases 3.0.0 added

#3 - 06/11/2021 09:08 AM - Adam Ruzicka
- Status changed from Ready For Testing to Closed

Applied in changeset foreman|150d024e6f45daa31bd8134380183dcb0e0736

#4 - 06/16/2021 01:53 PM - Amit Upadhye
- Target version set to 2.5.1

#5 - 06/23/2021 10:29 AM - Tomer Brisker
- Target version changed from 2.5.1 to 3.0.0

Looks like this potentially causes regressions in katello proxy handling, pushing out of 2.5.1 until we have a solution for Katello as well.