Bug Report 1

URL Validator fails to validate local URLs when 'ALLOW_LOCAL_URLS' is set

Submitter

Matt Walz

Date Seen

11 March 2016

Tested with following OS and software

Windows 10

Eclipse - Version: Kepler Service Release 2; Build id: 20140224-0627

Bug Description

When testing isValid() method for UrlValidator.java with UrlValidator.ALLOW_LOCAL_URLS set, the method incorrectly returns false for http://hostname and http://localhost.

Minimized test code to reproduce failure:

```
UrlValidator urlVal = new UrlValidator(null, null,
  UrlValidator.ALLOW_LOCAL_URLS);
assertTrue("http://hostname is valid",
  urlVal.isValid("http://hostname"));
assertTrue("http://localhost is valid",
  urlVal.isValid("http://localhost"));
```

Actual Behavior

Running the code listed above will result in both assertTrue checks failing.

Expected Behavior

I expected both assertTrue checks to pass on the above code with the UrlValidator.ALLOW_LOCAL_URLS' value set.

Troubleshooting/Testing Steps Attempted

I believe I have localized the error to the DomainValidator.java file, line number 139. The NOT (!) should be removed from the beginning of the 'if' statement. Thus, line 139 should read as:

```
if (hostnameRegex.isValid(domain)) {
```

Bug Report 2

URL Validator fails to invalidate IPv4 Addresses with byte values > 255

Submitter
Matt Walz

Date Seen
11 March 2016

Tested with following OS and software
Windows 10
Eclipse - Version: Kepler Service Release 2; Build id: 20140224-0627

Bug Description
When testing isValid() method for UrlValidator.java with UrlValidator.ALLOW_LOCAL_URLS set, the method incorrectly returns true for http://256.256.256.256.

Minimized test code to reproduce failure:

```
UrlValidator urlVal = new UrlValidator();
assertFalse("http://256.256.256.256 is invalid",
  urlVal.isValid("http://256.256.256.256"));
```

Actual Behavior
Running the code listed above will result in the assertFalse check failing.

Expected Behavior
I expected the assertFalse check to fail since the maximum integer value of a byte is 255.

Troubleshooting/Testing Steps Attempted
I believe I have localized the error to the inetAddressValidator.java file, line number 96. The 'return' statement after checking

    if (iIPSegment > 255)

should be a 'return false;' (instead of return true as is currently written).

Bug Report 3

URL Validator fails to validate many two-letter country codes used as internet top level domains

Submitter
Matt Walz

Date Seen
12 March 2016

Tested with following OS and software
Windows 10
Eclipse - Version: Kepler Service Release 2; Build id: 20140224-0627

Bug Description
When testing isValid() method for UrlValidator.java, URLs containing two-letter country codes beginning with the letter j or later (alphabetically speaking) as their domain codes are automatically invalidated.

Minimized test code to reproduce failure:

```
UrlValidator urlVal = new UrlValidator();
assertTrue("http://www.google.mz is valid",
  urlVal.isValid("http://www.google.mz"));
```

Actual Behavior
Running the code listed above will result in the assertTrue check failing.

Expected Behavior
I expected the assertTrue check to pass since ".mz" is the correct domain code for the country of Mozambique.

Troubleshooting/Testing Steps Attempted
I believe I have localized the error to the DomainValidator.java file. Beginning at line 250, an array containing strings of acceptable country codes is created, but the array stops with the string "it" for Italy. This array needs to be updated to contain the complete list of two-letter country domain codes.

Bug Report 4

URL Validator fails to validate valid port values > 999

Submitter

Matt Walz

Date Seen

12 March 2016

Tested with following OS and software

Windows 10

Eclipse - Version: Kepler Service Release 2; Build id: 20140224-0627

Bug Description

When testing isValid() method for UrlValidator.java, URLs containing port values greater than 999 are automatically invalidated.

Minimized test code to reproduce failure:

```
UrlValidator urlVal = new UrlValidator();
assertTrue("http://www.google.com:1023 is valid",
  urlVal.isValid("http://www.google.com:1023"));
```

Actual Behavior

Running the code listed above will result in the assertTrue check failing.

Expected Behavior

I expected the assertTrue check to pass since :1023 is a valid port value (last of the well-known port numbers)

Troubleshooting/Testing Steps Attempted

I believe I have localized the error to the URLValidator.java file, line 158. This line creates a regular expression for ports which is limited to 3 digits. If this was changed to allow 5 digits it would allow the validator to account for all well-known, registered, and dynamic/private port numbers. However a check would still be needed to verify that the port value is less than or equal to 65535 as that is the highest allowed dynamic port value.

Bug Report 5

URL Validator fails to validate URLs with queries

Submitter

Matt Walz

Date Seen

12 March 2016

Tested with following OS and software

Windows 10

Eclipse - Version: Kepler Service Release 2; Build id: 20140224-0627

Bug Description

When testing isValid() method for UrlValidator.java, URLs containing query strings are automatically invalidated.

Minimized test code to reproduce failure:

```
UrlValidator urlVal = new UrlValidator();
assertTrue("http://www.google.com?query=answer",
  urlVal.isValid("http://www.google.com?query=answer"));
```

Actual Behavior

Running the code listed above will result in the assertTrue check failing.

Expected Behavior

I expected the assertTrue check to pass since "`?query=answer`" is a valid query string.

Troubleshooting/Testing Steps Attempted

I believe I have localized the error to the URLValidator.java file, line 446. The NOT (!) should be removed from the beginning of the 'return' statement. Thus, line 446 should read as:

return QUERY_PATTERN.matcher(query).matches();