**Problem Statement:**

A2019 | For mail trigger bots: read mail is returned to unread.

**Problem Analysis:**

■ Procedure

Client set a bot with email trigger, deployed to BR.

She realized that when Email Event Trigger fired, one of the read emails has been returned to unread.

Also, it continued to occur even after deleting the corresponding bot from "Activity"-"Event Trigger".

To eliminate this event, the BR user needs to sign out of the device.

The mail server set as the trigger is imap.gmail.com

There was no record in "Activity"-"History" or "Audit Log".

■ Error details

N/A

■ Others

N/A

**When did it start happening:**

After Email Trigger “Check every:” value was modified.

**Did anything change before it started:**

“Check every:” value of Email Trigger was modified.

**How many Users/VDIs are affected:**

All

**Any existing KB shared/referred:**

No

**Logs Collected & reviewed:**

Attached my Node\_Manager.log, Bot\_launcher.log and Trigger\_Listener.log files

**Findings from the review of logs:**

[Node\_manager.log]

2020-Nov-30 Mon 14:05:34.515 INFO [pool-10-thread-1] - com.automationanywhere.nodemanager.service.impl.message.handlers.TriggerListenHandler - {} - process(TriggerListenHandler.java:86) - Successfully sent message to TRIGGER\_LISTENER\_START and payload:{ "triggerConfigurations": [{ "uid": "74b05dc6-f2ab-4c00-a813-c4d731c223ad", "triggerName": "EmailTrigger", "attributes": [{ "id": "", "name": "serverType", "value": { "type": "STRING", "expression": "", "string": "EMAIL\_SERVER", "number": "", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "serverHost", "value": { "type": "STRING", "expression": "", "string": "imap.gmail.com", "number": "", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "protocol", "value": { "type": "STRING", "expression": "", "string": "IMAP", "number": "", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "useSecure", "value": { "type": "BOOLEAN", "expression": "", "string": "", "number": "", "boolean": "true", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "port", "value": { "type": "NUMBER", "expression": "", "string": "", "number": "993", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "pollIntervalForEmail", "value": { "type": "NUMBER", "expression": "", "string": "", "number": "15", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "unitForEmail", "value": { "type": "STRING", "expression": "", "string": "SECONDS", "number": "", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "username", "value": { "type": "STRING", "expression": "", "string": "aileeli921@gmail.com", "number": "", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "securedPassword", "value": { "type": "STRING", "expression": "", "string": "Ly+3626538", "number": "", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "emailType", "value": { "type": "STRING", "expression": "", "string": "INCOMINGS", "number": "", "boolean": "", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "checkForSender", "value": { "type": "BOOLEAN", "expression": "", "string": "", "number": "", "boolean": "false", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "checkForSubject", "value": { "type": "BOOLEAN", "expression": "", "string": "", "number": "", "boolean": "false", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "checkForEmailWithoutAttachement", "value": { "type": "BOOLEAN", "expression": "", "string": "", "number": "", "boolean": "false", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "checkForEmailWithAttachement", "value": { "type": "BOOLEAN", "expression": "", "string": "", "number": "", "boolean": "false", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }, { "id": "", "name": "checkForFolder", "value": { "type": "BOOLEAN", "expression": "", "string": "", "number": "", "boolean": "false", "repository": "", "variableName": "", "packageId": "", "packageName": "", "iteratorId": "", "iteratorName": "", "conditionalId": "", "conditionalName": "", "list": [], "dictionary": [], "exceptionName": "", "thumbnailMetadataPath": "", "screenshotMetadataPath": "", "properties": [], "triggerName": "", "formElementId": "", "formElementType": "", "formInstanceId": "", "queueName": "", "taskbotSelf": false }, "attributes": [], "operator": "NONE" }], "triggerPackage": { "downloadUrl": "C:\\ProgramData\\AutomationAnywhere\\GlobalCache\\zTn4+HreanLjIYs2wvsTNag65kRAIDOOzK1yTr+jUs\u003d\\bot-trigger-email-1.1.0-20200903-114514.jar", "uri": "https://aa-support-v2-iq-jp-2.my.automationanywhere.digital/v2/packages/artifacts/resource/bot-trigger-email-1.1.0-20200903-114514", "dependencyType": "TRIGGER\_PACKAGE" }, "resources": [], "botId": "0", "triggerDataVariable": "" }], "deviceUserName": "CSSDALLAP01199\\sysadmin", "id": "b6dd7f2a-3835-43a3-829b-df2372ce4efd", "userToken": "eyJhbGciOiJSUzUxMiJ9.eyJzdWIiOiI1MDgiLCJjbGllbnRUeXBlIjoiQUFFX0NMSUVOVCIsInRlbmFudFV1aWQiOiJmYTlkZjg2ZC0yNmUyLTQ5NjktYTZlMy04NzhkOWQ2OWUyYzAiLCJoeWJyaWRUZW5hbnQiOiIiLCJpYXQiOjE2MDY3MTI3MzQsImV4cCI6MTYwNjcxMzkzNCwiaXNzIjoiQXV0b21hdGlvbkFueXdoZXJlIiwibmFub1RpbWUiOjI0NjU2NzcxODU3NTE3Mzh9.M3aZcqSP7jndSg-vk3GLaduoS9rnMXx7Wa7khxTGwdnf8gtrPDRBC9R2t\_NlzrszqNk0aa79mMFYvWNWCQsMpk6SjIduHuundwM1V8QfMpuaLPCwVVH9qdU0A62y8P2RAnp6g7hEW\_rKIzshXSk7pZsCmXfksaVhCy1g5Mm3MILqjgKNA67MOYT2-yUtyIRKN2hFPbLNiB07lKoJgMqrQ4QEUGYA5yrQMqO6YoVXHhgcu0sitaFI1WwXdcm842-dy1sGcRg-\_1am6WXCh6wN4b0uwFhGhVQhVyCPZ23Qfq84oW6tttJRVKDGOwwCLjoET4YmK4CepNTkPCM59AjUiQ", "crUrl": "https://aa-support-v2-iq-jp-2.my.automationanywhere.digital", "proxyUser": "", "proxyPassword": "", "proxyInitVector": "", "deviceType": "SINGLE\_USER"}

=================

[Trigger\_Listener.log]

2020-Nov-30 Mon 14:05:26.999 DEBUG [Timer-0] - com.automationanywhere.botcommand.email.commands.EmailConnect - execute - sessionName:default, serverHost:imap.gmail.com, port:993.0,useSecure:true, protocol:IMAP

2020-Nov-30 Mon 14:05:26.999 INFO [Timer-0] - com.automationanywhere.botcommand.email.ck.LibararyLoader - loadDlls - dlls are loaded from location:/libs/chilkat.dll , extracted absulute path:

2020-Nov-30 Mon 14:05:26.999 DEBUG [Timer-0] - com.automationanywhere.botcommand.email.factory.ServiceProviderFactory - getEmailReceiverClient - Created Client type of com.automationanywhere.botcommand.email.ck.CkImapReaderClient@174d314 for protocol IMAP

2020-Nov-30 Mon 14:05:26.999 DEBUG [Timer-0] - com.automationanywhere.botcommand.email.ck.CkImapReaderClient - getCkImap - CkImap object is created

**Steps Performed:**

(1) Created a Bot with Email Trigger. (Trigger setting is as below: )

Connect to Email: Email server

Host: imap.gmail.com

Protocol for retrieving message: IMAP

Use secure connection(SSL/TLS): Yes

Port: 993

Check every: 20 Seconds

Username: (confidential)

Password: (confidential)

Emails that will start the bot: All incoming emails

Select Conditions: (None)

(2) Checked the Bot in. Deployed the Event Trigger to BR.

(3) Make sure all mails in my Gmail inbox was read. Send a test mail to my Gmail address to fire the Email Trigger.

(4) No read mail changed into un-read, and bot was triggered successfully.

(5) Checked the Bot out. Modified the Email trigger value of “Check every:” (from “20 Seconds” into “15 Seconds”.) Save. Checked the Bot in. Deployed the Event Trigger to BR again.

(6) Mark all mails in my Gmail inbox as read. Send a test mail to my Gmail address to fire the Email Trigger.

(7) Received test mail. At the meantime, the read mail below test mail was changed into un-read. Bot was triggered successfully.