

Overview

HornetQ Changes to create a twitter bridge (Nice To Have)

<https://jira.jboss.org/jira/browse/HORNETQ-189>

Description From Jira

Create a twitter module to allow tweets to be consumed in HornetQ and sent to an address.

Also consider outgoing tweets - allow messages to be consumed from a queue (like a bridge) and forwarded to twitter

There is a module here:

<http://pilhuhn.blogspot.com/2009/08/twitter-plugin-alert-sender-for-jopr.html>

Tim Fox - 20/Jan/10 07:33 AM

This task could be done in two parts

1) Basically the idea here is it will be possible to configure a HornetQServer to log into twitter as a particular user and receive tweets, those tweets would then be sent to a configurable address in HornetQ as messages.

2) Also we could consider outgoing tweets. I.e. HornetQ can send tweets to twitter by consuming messages from a queue and forwarding them.

References

Twitter API Wiki

<http://apiwiki.twitter.com>

OAuth

<http://hueniverse.com/oauth/>

Twitter API Documentation - REST APIs

<http://apiwiki.twitter.com/Twitter-API-Documentation>

Twitter API Documentation - Streaming

<http://apiwiki.twitter.com/Streaming-API-Documentation>

Design

Key Points

1. This functionality will be developed as a Protocol and will provide a new Protocol Type for HornetQ alongside the Stomp protocol.

2. The Protocol will be split into two pieces.
 1. An inbound component to allow HornetQ to receive Twitter status updates on a queue or topic.
 2. An outbound component to allow a Queue or Topic to be bridged to Twitter. This means messages posted to this queue/topic get posted as updates on Twitter for the specified account.
3. Twitter has 3 APIs. Two REST APIs and one Streaming API. Currently you can only post updates through the REST api. You can use either API to receive status updates though. The Protocol will be configurable as to which API is used for receiving status updates.
4. As the Twitter REST APIs are rate limited the components using the REST api will be configured with Rate Limiting by default. With the ability to turn off the rate limiting through configuration. The limits will be cumulative across both the Inbound and the Outbound components that use the same account. The limits as provided by Twitter are:
 1. 1,000 total updates per day, on any and all devices (web, mobile web, phone, API, etc.)
 2. 250 total direct messages per day, on any and all devices
 3. 150 API requests per hour
5. The Streaming API is not rate limited so there is no requirement to throttle the Streaming API.
6. The Twitter REST apis support both OAuth and basic authentication. The Outbound component and the Inbound REST component will be configurable for either authentication method. The Streaming API only supports basic authentication so there will be no alternative authentication configuration for the Inbound Streaming component.

Plan Of Attack

Option 1 - Third Party Library

Use a third party library for communicating with Twitter. For example Twitter 4J (<http://twitter4j.org/en/index.html>)

Even with the use of a third party library we still need to do some dev work.

1. Implement a Rest API solution for connecting a HornetQ Queue or Topic to Twitter to receive Updates.
2. Implement a Streaming API solution for connecting a HornetQ Queue or Topic to Twitter to receive Updates.
3. Implement a Rest API solution for bridging a HornetQ Queue or Topic to Twitter so messages placed on the queue are sent as updates to Twitter.

It is just that the solutions would hand off the communicating to Twitter to the third party library.

Option 2 -Roll Your Own.

1. Implement support for Basic Authentication.
 1. I don't know if HornetQ already has this functionality implemented
 2. Twitter is phasing it out but the Streaming API only supports basic authentication.

2. Implement support for OAuth.
 1. I don't know if HornetQ already has OAuth support so have added it as a task here.
3. Implement a Rest API solution for connecting a HornetQ Queue or Topic to Twitter to receive Updates.
4. Implement a Streaming API solution for connecting a HornetQ Queue or Topic to Twitter to receive Updates.
5. Implement a Rest API solution for bridging a HornetQ Queue or Topic to Twitter so messages placed on the queue are sent as updates to Twitter.