

Robin MTA Tester



About

As an engineer at Mimecast working on the front line team that's responsible for our Mail Transfer Agent it is my responsibility to debug, maintain and update the various software and libraries handling our email receipt and delivery capabilities.

The MTA is the way in and out for all emails and when processing billions of emails every week we need to be as accepting as possible in order to ensure our clients get their emails regardless of any differences in protocol implementations and interpretations that exist.

Given the multitude of email client libraries available in a variety of languages and frameworks in use in the ever evolving SMTP ecosystem we have encountered many email receipt issues due to non RFC compliant clients and over time we found a need to test and ensure our MTA systems are robust, scalable and secure in any scenario to maintain our highly demanding SLA's.

In doing this review it became apparent that we needed a more versatile client that is capable of reproducing edge and negative test case behaviours to meet our testing needs.

At first there was a shell script; that slowly evolved into a small Java library; and over time into a large repository of test cases used by our engineers in their daily

development activities.

Now we have reached a level of maturity and value that we feel it is a good time to share this with the OpenSource community in the hope that it will help others in their testing efforts, as well as leading towards a more secure global email environment for us all.

Purpose

The primary purpose of this tool is testing and as such we tried to make it as language agnostic as possible so it can be easily used by any engineer regardless of their knowledge of Java.

In order to remain agnostic we have used JSON files to enable users to configure test “cases”. By using the simple CLI interface, anyone can run tests with ease.

You’ll note that the client and server components are not hardened and can be configured to exhibit bad behaviours unlike most available libraries — this is by design, as we use this tool for negative as well as positive test cases.

Robin is now powering all our email test suites running tens of thousands of tests in all possible configurations to ensure our MTA’s stay highly robust.

While knowing Java is not required; it is beneficial as due to our use of JUnit 5 and Maven to power the framework, engineers with an understanding of Java, JUnit and Maven can design suites of tests that can be used in more elaborate automation suites for testing MTA behaviours.

Also, an interface for a logs client is provided, which can be used to fetch the logs of the target MTA for added assertion capabilities in your testing.

War story

As it happens sometimes in development you run into mind bending problems that do not seem to make any sense. You look at all the known variables and they don’t add up.

This was the case some time ago when we seemingly stopped receiving the required amount of bytes advertised by the BDAT command from one specific ESP alone.

We started by adding extra logging in the hope this would help clear things up, but this was not the case. We tried to reproduce the issue by fiddling with how the bytes are written to the socket, yet to no avail.

It was time for extreme measures, so we took a development server and disabled TLS support and added packet tracing so we could inspect what was happening at protocol level.

This was when we noticed that this specific ESP was sending the BDAT SMTP command and the first few hundred bytes of the email in a single write.

We implemented this into Robin so we can reproduce the scenario and success, we knew how to cause the issues, now it was time to figure out why we were not accounting for those bytes and it didn't take long before we had a solution.

Why Robin?

I have a Robin nest in my garden and as I was watching them peck at the feeder I thought that it seemed quite a fitting mascot for a tool that pecks at an MTA. Much like Robin Hood, if he was thought of as a castle defences tester.