**08/03/2024 Agilysys: Requirement to use Sendgrid API with scheduler configuration**

*—----EMAIL FROM THE CUSTOMER—----*

*Hi Mehul,*

*We would like to utilize the Jasper Scheduler for reports, As per our understanding, we need to configure the below details to enable it but we are using sendGrid for scheduling and send email, we are using API Key based authentication and port 25 we can’t use. Could you please verify Jasper will support the API Key?*

*# quartz.mail.sender.host=*[*mail.example.com*](http://mail.example.com/)

*# quartz.mail.sender.port=25 : we can use 587*

*# quartz.mail.sender.protocol=smtp*

*# quartz.mail.sender.username=admin*

*# quartz.mail.sender.password=password*

*# quartz.mail.sender.from=admin@example.com*

*—----EMAIL FROM THE CUSTOMER END—----*

Agilysys would like to know if its possible to use Sendgrid to send emails from the scheduler using the API keys

They specifically need to configure API keys for authentication and do not want to use the SMTP option with username//password.

<https://docs.sendgrid.com/for-developers>

<https://sendgrid.com/en-us/blog/web-api-or-smtp-relay-how-should-you-send-your-mail>

I have told the customer that there is no out of the box support for API key authentication out of and that it would need to be scoped and customized. We are not sure if the customer is going to use the templates from sendgrid or not yet.

**Possible Solution?**

Can we have a custom **reportSchedulerMailSender** instead of the one below **JavaMailSenderImpl** from Spring and inject that into the scheduler service?

<bean id="**reportSchedulerMailSender**" class="org.springframework.mail.javamail.**JavaMailSenderImpl**" lazy-init="true">

<property name="host" value="${report.scheduler.mail.sender.host}"/>

<property name="username" value="${report.scheduler.mail.sender.username}"/>

<property name="password" value="${report.scheduler.mail.sender.password}"/>

<property name="protocol" value="${report.scheduler.mail.sender.protocol}"/>

<property name="port" value="${report.scheduler.mail.sender.port}"/>

<property name="javaMailProperties">

<props>

<prop key="mail.smtp.sendpartial">true</prop>

<prop key="mail.smtp.auth">false</prop>

</props>

</property>

</bean>

The custom class would then handle the send and other required functions?

Sendgrid Java library:

<https://github.com/sendgrid/sendgrid-java/>

**Questions for Engineering:**

1. Any recommendations or alternative approach?
2. Validation of the approach?
3. Any impacts on JRS Scheduling?
4. Efforts from a development perspective?