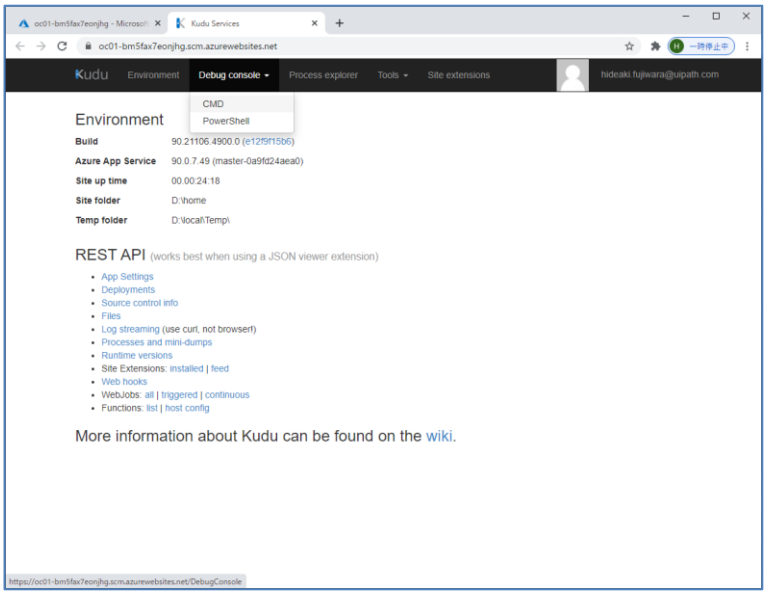
Azure App Service における UiPath Orchestrator イベントログ出力

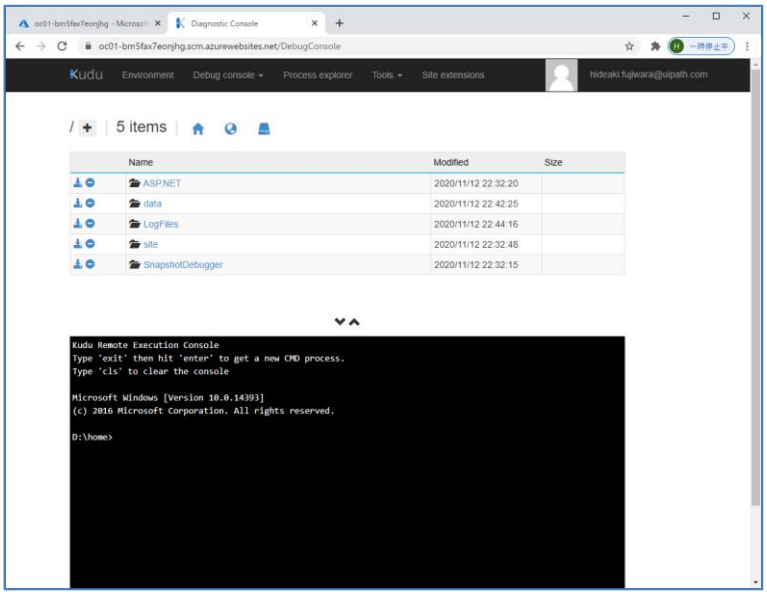
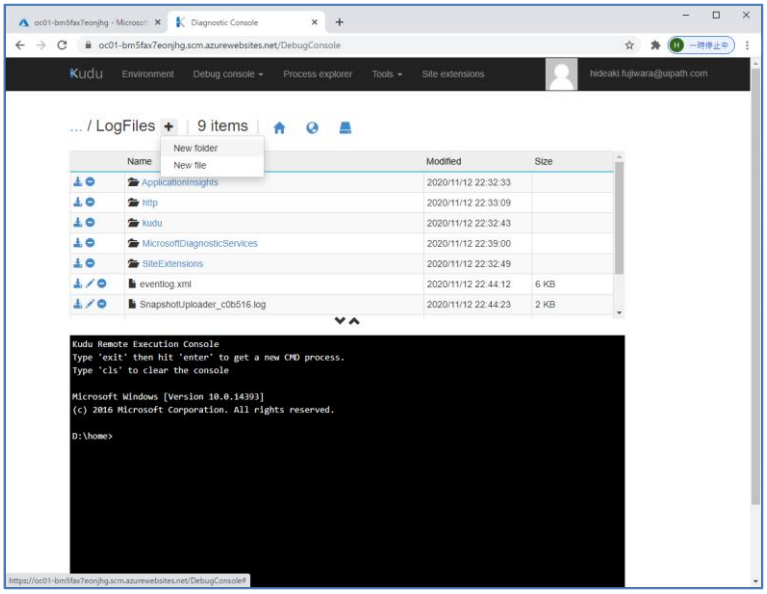
UiPath Robotic Process Automation ®

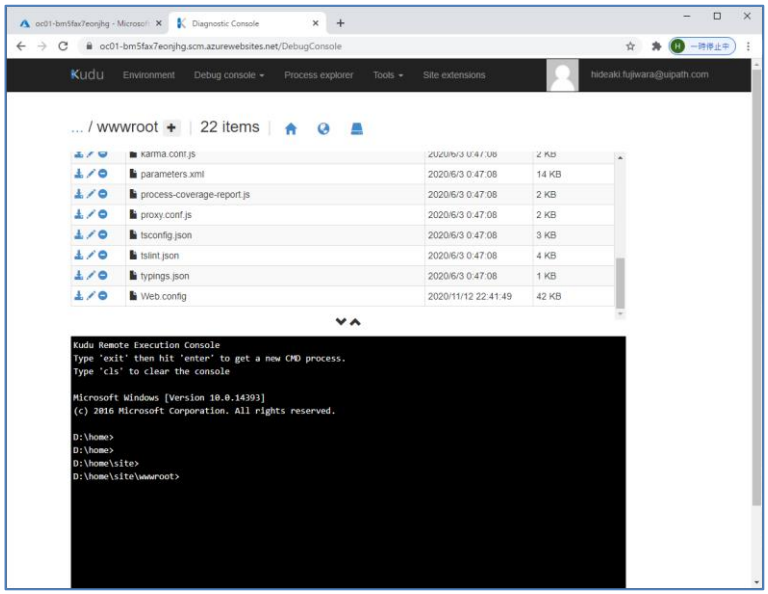
Revision History

Date	Version	Author	Description
12/10/2020	1.0	UiPath Japan Infra Team	First version

本文書では Azure App Service 上に構築された UiPath Orchestrator のアプリケーションイベントログをテキストファイルに日付ごとに出力する手順について説明します。

	Orchestrator の App Service において、メニューより「高度なツール」(Kudu) を起動
	Debug Console > CMD を起動

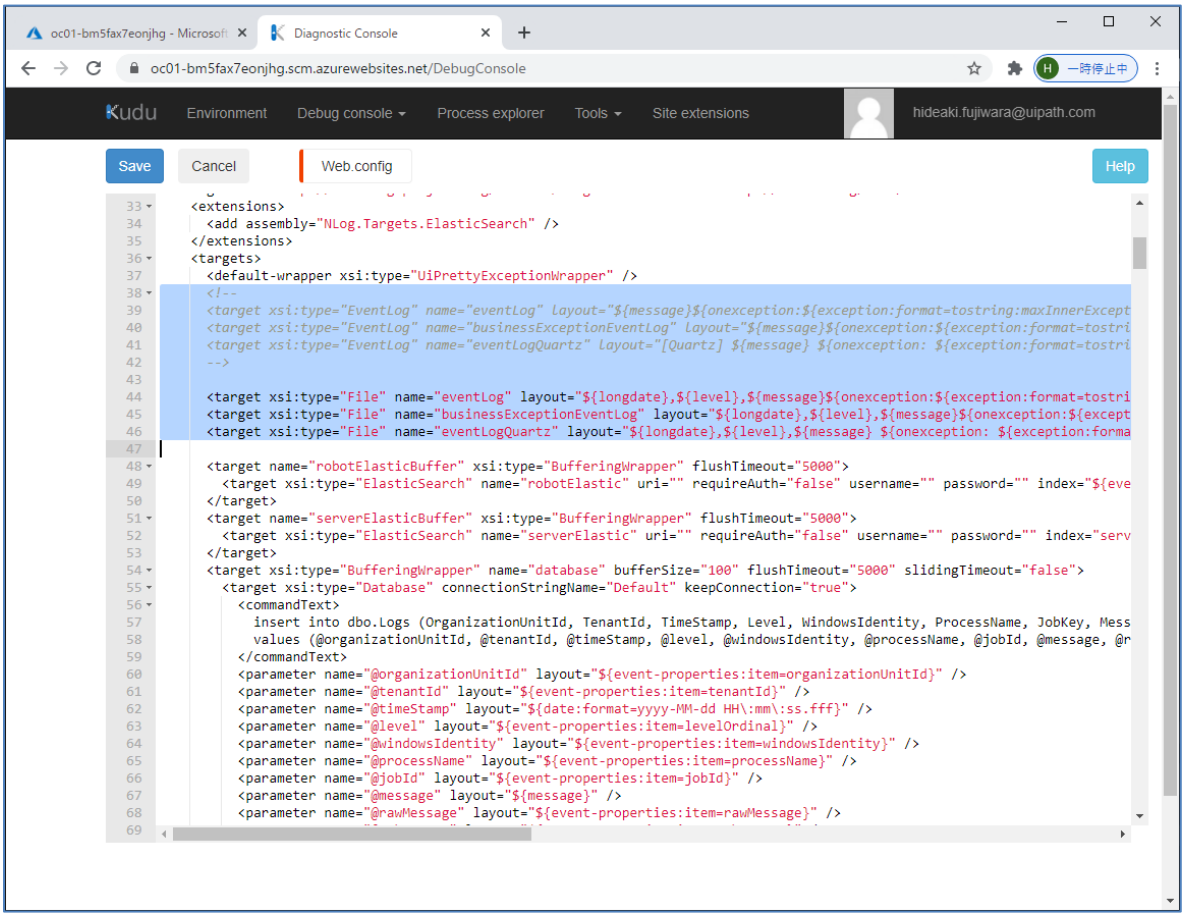
	上部ファイルエクスプローラーにおいて LogFiles をクリック
	+ をクリックし、New folder → Orchestrator という名前のフォルダーを作成

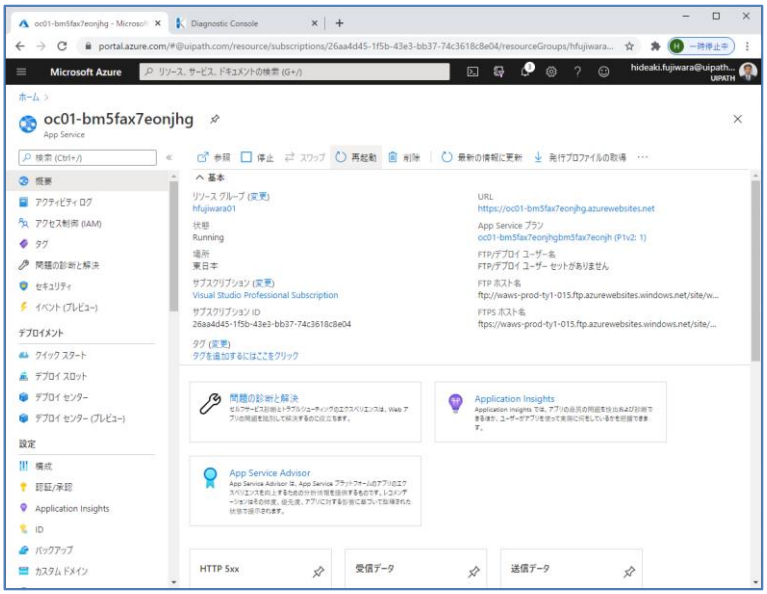
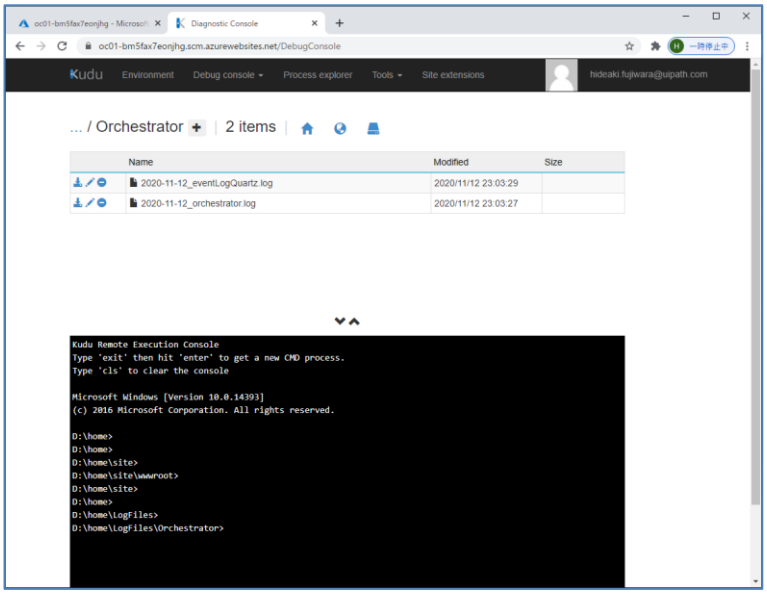
	D:\home\site\wwwroot まで移動し、Web.config (v2020.10 以降では UiPath.Orchestrator.dll.config) を編集

下記の 3 行をコメントアウトし、3 行を追加し、Save する。

```
<!--
<target xsi:type="EventLog" name="eventLog"
layout="${message}${onexception:${exception:format=tostring:maxInnerExceptionLevel=5:innerFormat=tostring
}}" source="Orchestrator" log="Application" />
<target xsi:type="EventLog" name="businessExceptionEventLog"
layout="${message}${onexception:${exception:format=tostring:maxInnerExceptionLevel=5:innerFormat=tostring
}}" source="Orchestrator.BusinessException" log="Application" />
<target xsi:type="EventLog" name="eventLogQuartz" layout="[Quartz] ${message} ${onexception:
${exception:format=tostring:maxInnerExceptionLevel=5:innerFormat=tostring}}" source="Orchestrator"
log="Application" />
-->

<target xsi:type="File" name="eventLog"
layout="${longdate},${level},${message}${onexception:${exception:format=tostring:maxInnerExceptionLevel=5:i
nnerFormat=tostring}}" fileName="D:/home/LogFiles/Orchestrator/${shortdate}_orchestrator.log"
keepFileOpen="true" encoding="utf-8" />
<target xsi:type="File" name="businessExceptionEventLog"
layout="${longdate},${level},${message}${onexception:${exception:format=tostring:maxInnerExceptionLevel=5:i
nnerFormat=tostring}}" fileName="D:/home/LogFiles/Orchestrator/${shortdate}_businessException.log"
keepFileOpen="true" encoding="utf-8" />
<target xsi:type="File" name="eventLogQuartz" layout="${longdate},${level},${message} ${onexception:
${exception:format=tostring:maxInnerExceptionLevel=5:innerFormat=tostring}}"
fileName="D:/home/LogFiles/Orchestrator/${shortdate}_eventLogQuartz.log" keepFileOpen="true"
encoding="utf-8" />
```


oc01-bm5fax7eonjhg - Microsoft
Diagnostic Console
oc01-bm5fax7eonjhg.scm.azurewebsites.net/DebugConsole
一時停止中
Kudu
Environment
Debug console
Process explorer
Tools
Site extensions
hideaki.fujiwara@uipath.com
Save
Cancel
Web.config
Help
<extensions>
<add assembly="NLog.Targets.ElasticSearch" />
</extensions>
<targets>
<default-wrapper xsi:type="UiPrettyExceptionWrapper" />
<!--
<target xsi:type="EventLog" name="eventLog" layout="${message}${onexception:${exception:format=tostring:maxInnerExcept
<target xsi:type="EventLog" name="businessExceptionEventLog" layout="${message}${onexception:${exception:format=tostri
<target xsi:type="EventLog" name="eventLogQuartz" layout="[Quartz] ${message} ${onexception: ${exception:format=tostri
-->
<target xsi:type="File" name="eventLog" layout="${longdate},${level},${message}${onexception:${exception:format=tostri
<target xsi:type="File" name="businessExceptionEventLog" layout="${longdate},${level},${message}${onexception:${except
<target xsi:type="File" name="eventLogQuartz" layout="${longdate},${level},${message} ${onexception: ${exception:forma
<target name="robotElasticBuffer" xsi:type="BufferingWrapper" flushTimeout="5000">
<target xsi:type="ElasticSearch" name="robotElastic" uri="" requireAuth="false" username="" password="" index="${eve
</target>
<target name="serverElasticBuffer" xsi:type="BufferingWrapper" flushTimeout="5000">
<target xsi:type="ElasticSearch" name="serverElastic" uri="" requireAuth="false" username="" password="" index="serv
</target>
<target xsi:type="BufferingWrapper" name="database" bufferSize="100" flushTimeout="5000" slidingTimeout="false">
<target xsi:type="Database" connectionStringName="Default" keepConnection="true">
<commandText>
insert into dbo.Logs (OrganizationUnitId, TenantId, TimeStamp, Level, WindowsIdentity, ProcessName, JobKey, Mess
values (@organizationUnitId, @tenantId, @timeStamp, @level, @windowsIdentity, @processName, @jobId, @message, @r
</commandText>
<parameter name="@organizationUnitId" layout="${event-properties:item=organizationUnitId}" />
<parameter name="@tenantId" layout="${event-properties:item=tenantId}" />
<parameter name="@timeStamp" layout="${date:format=yyyy-MM-dd HH\:mm\:ss.fff}" />
<parameter name="@level" layout="${event-properties:item=levelOrdinal}" />
<parameter name="@windowsIdentity" layout="${event-properties:item=windowsIdentity}" />
<parameter name="@processName" layout="${event-properties:item=processName}" />
<parameter name="@jobId" layout="${event-properties:item=jobId}" />
<parameter name="@message" layout="${message}" />
<parameter name="@rawMessage" layout="${event-properties:item=rawMessage}" />
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69

	Orchestrator の App Service 概要に戻り、再起動
	Kudu で D:\home\LogFiles\Orchestrator に移動し、イベントログがファイル出力されていることを確認。

以上