encamina
PIENSA EN COLORES

How to generate Let's Encrypt SSL Certificates with Azure Automation

Autor Dani Alonso

Thanks to **Let's Encrypt** we can have as many SSL certificates as we need, for free and for life. The only drawback is that these expire every 3 months, which could easily make us back down... Luckily, thanks to the magic of **PowerShell** and **Azure Automation** we can set up the default renewal schedule and take our minds off this task. And the best part is that it is fully **compatible with any Azure Web Apps**!

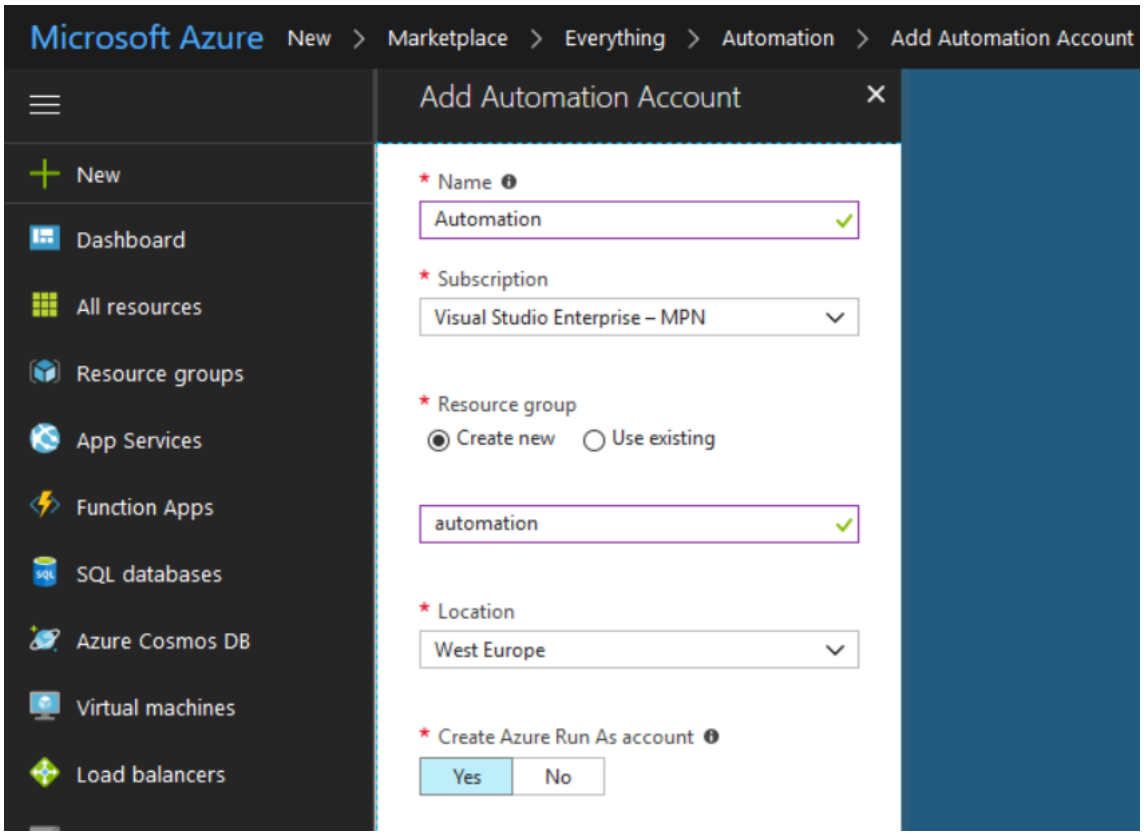
First of all, we need to have an **Azure Automation** account in our Azure subscription. If you already have on, you can skip the next step.

1. Add an Azure Automation Account

As I mentioned in the preceding paragraph, you may skip this step if you already have an Azure Automation account in your subscription. Otherwise, go ahead and follow these three simple steps:

1. Log into the Microsoft Azure portal

2. You can use the search box to find “Automation account”
3. Add a name, subscription, resource group and location. As you can see on the picture below, I’ve used “automation” both for the name and the resource group field

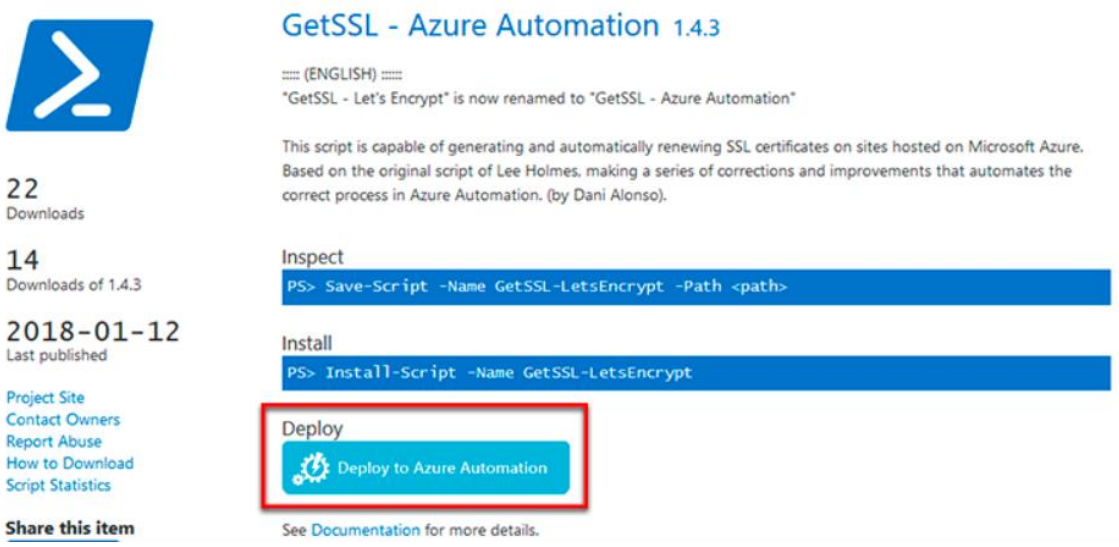


Then wait until the implementations that you have started are completed, and you will have at your disposal an Azure Automation account and we’ll be ready to move on.

2. Deploy GetSSL-LetsEncrypt to Azure

This is also a very simple process. To install the script on Azure Automation, click the following link of the **PowerShell Gallery**:

https://www.powershellgallery.com/packages/GetSSL-LetsEncrypt/1.4.3/



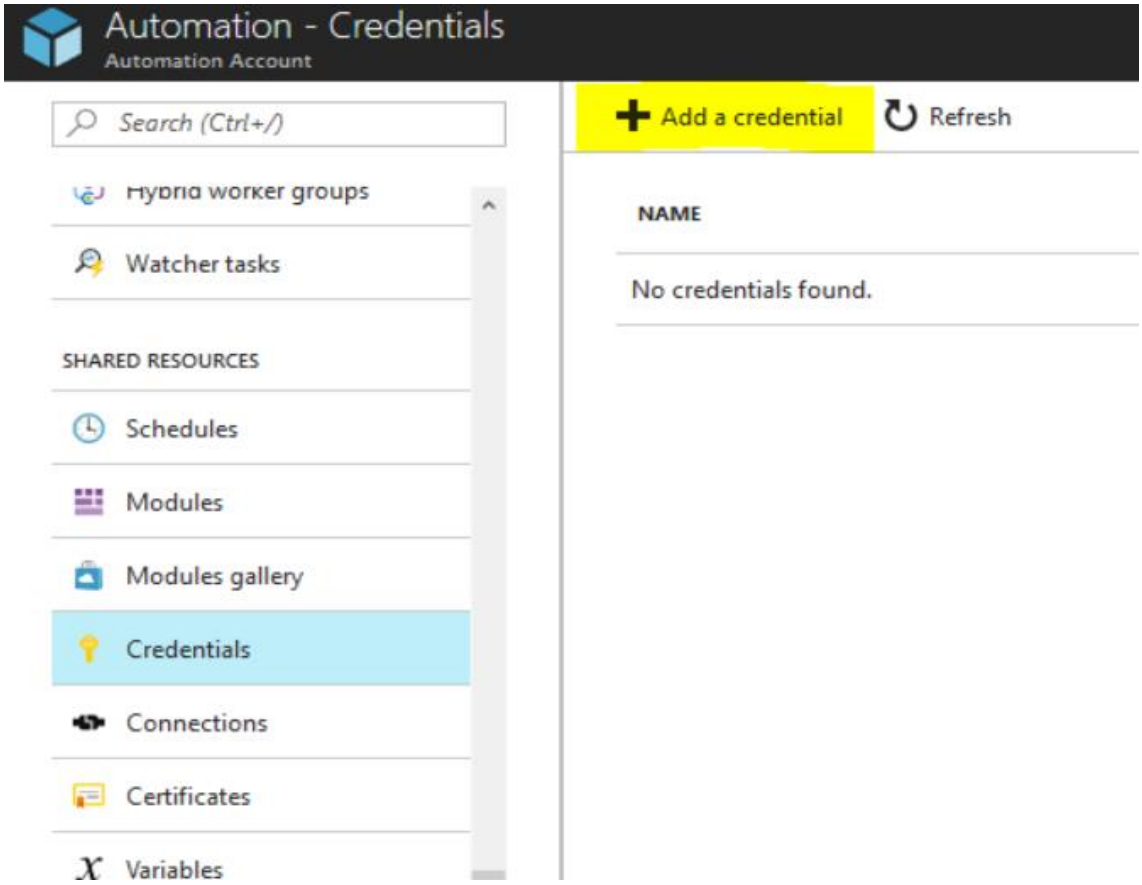
Then, click on "**Deploy to Azure Automation**", and the imported Runbooks will automatically open in Azure. Now, select your Automation account and accept.

As soon as the implementation begins, the installation of the modules that GetSSL-LetsEncrypt needs to operate will start. If you don't have the required modules, this implementation process can take about 5 minutes to complete. However, if you already have the modules installed, the process will be instantaneous. In any case, it will be carried out automatically, and without the need for regular monitoring.

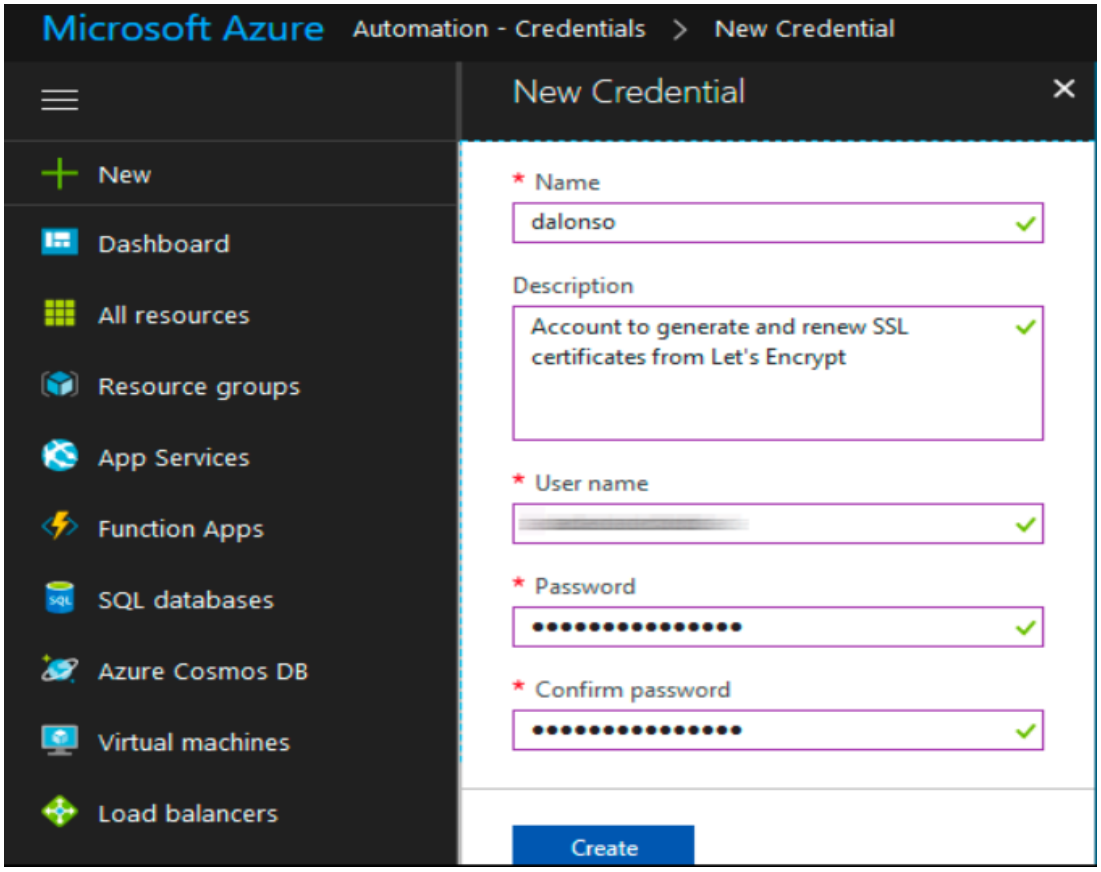
3. GetSSL Let's Encrypt Implementation

Once the GetSSL-LetsEncrypt deployment is completed, we have to set up a credential to automate the processes.

1. Go to Automation Account > Shared Resources > Credentials. Then click on "+ Add a credential"

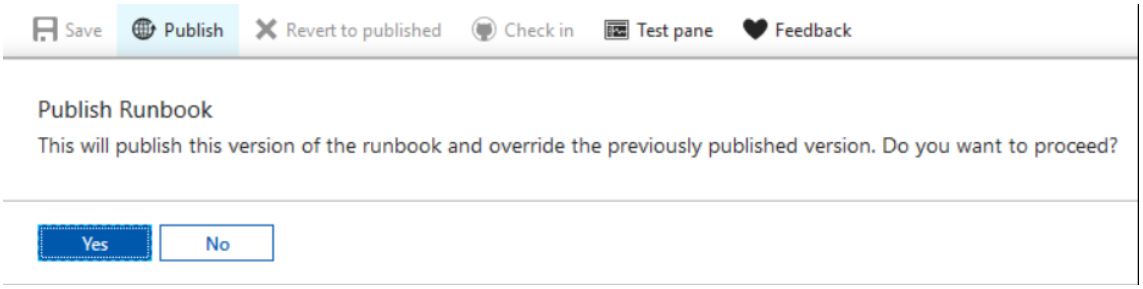


2. Create the account and specify the user account and password that you use to access the Azure Portal. For the example below, I've used "dalonso"



3. Now go to **Process Automation** > **Runbooks**. There you'll see the GetSSL-LetsEncrypt Runbook. Click on it

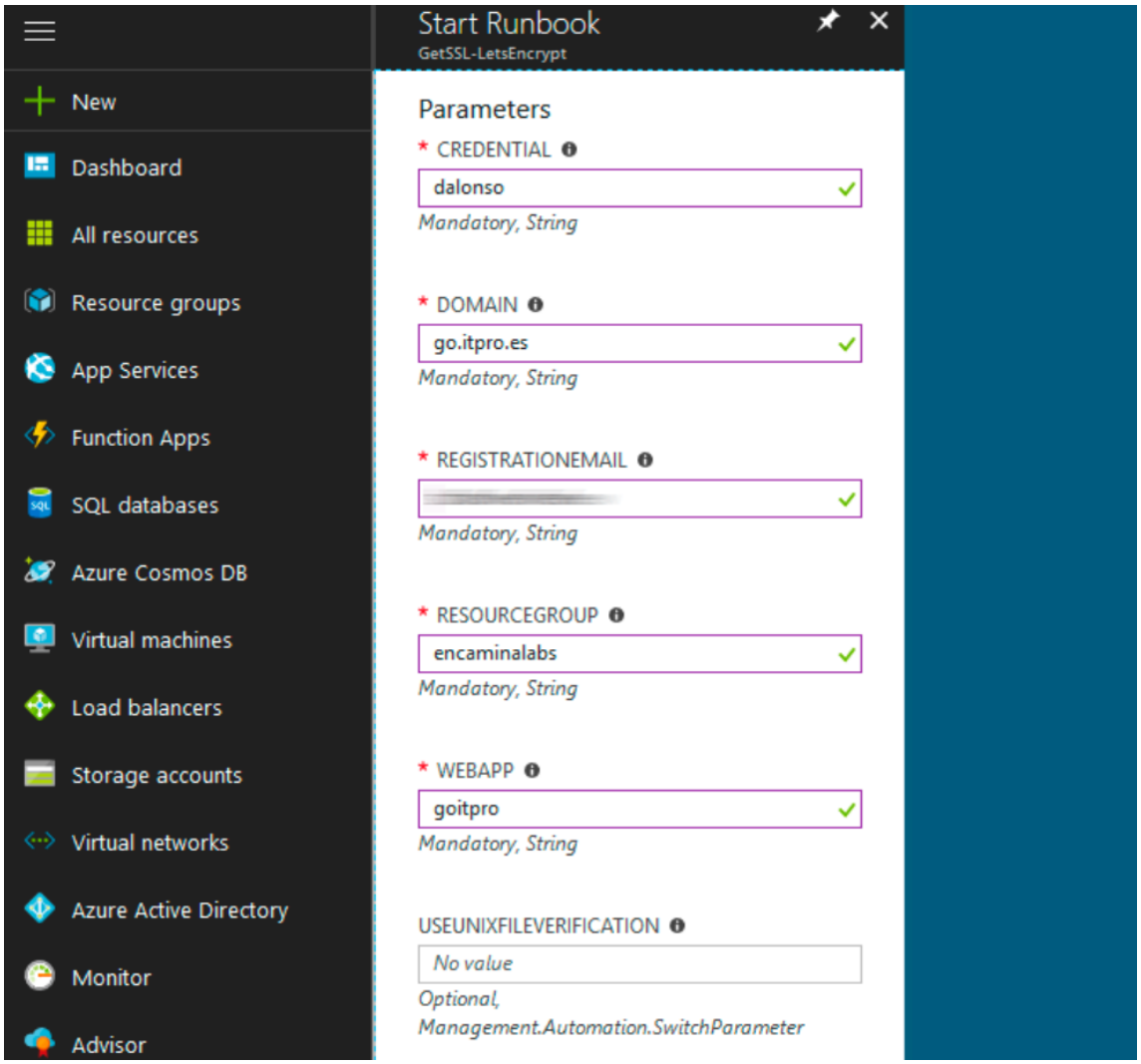
4. At the top, you will see various buttons. Click on Edit, and on the next page, click directly on **Publish**, answering **YES** to the question you see below



Before scheduling the task it is very important to make sure that everything works properly. Once you've done this, click on **Start** and fill in the required parameters:

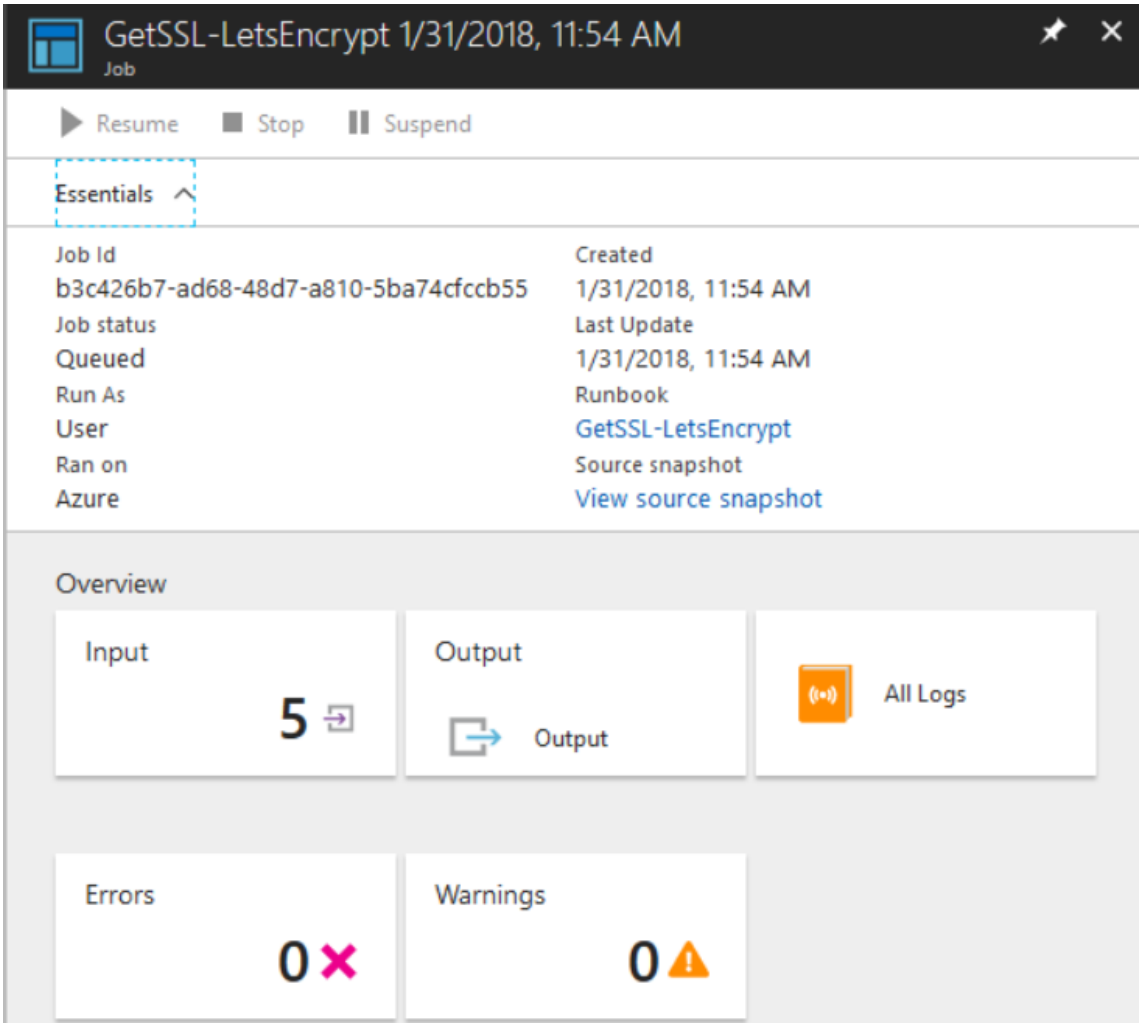
- **Credential**: Specify the credential account that you have created. In my case, that would be dalonso
- **Domain**: The domain where you want to add the certificate

- **Registrationemail**: The email where you want to receive notifications (e.g. expiration notices) from Let's Encrypt
- **Resourcesgroup**: Resource group
- **Webapp**: Name of the web instance
- **UneUnixFileVerification**: This is an optional and commonly unnecessary value



Fill it out and click **OK**.

When you're done, check and make sure that no errors have occurred. If everything is correct, open up the browser and load the website where you are installing the SSL certificate…



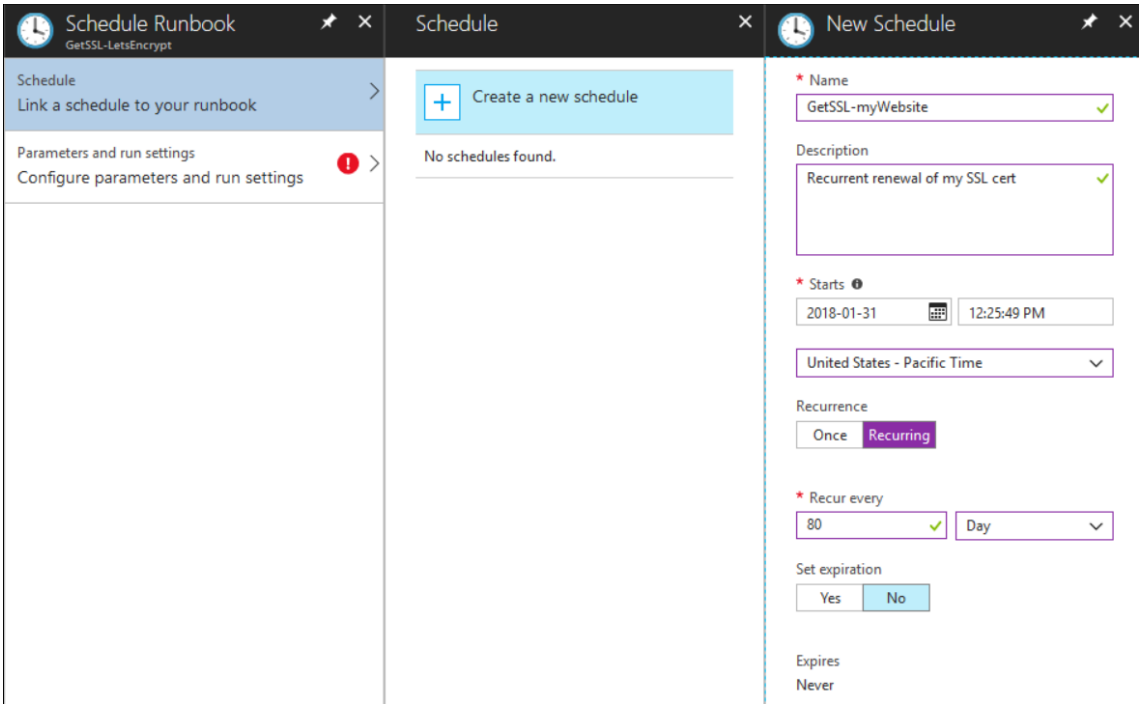
Oh yeah! **Your website is now working with SSL!** Let's set up the periodic renewal process.

4. Set up Automatic Renewal

This is it! Let's configure a scheduled task to run the script every once in a while so that we can forget about having to renew the certificate every three months.

1. Access the GetSSL-LetsEncrypt Runbook, and click on **Schedule**
2. Then click on **Scheduling** > **Create a new schedule**
3. Fill out all the scheduling data, enter a name as descriptive as possible, a brief description, a start date, etc. Keep in mind that the certificates have a validity period of

3 months, so **an 80 days periodicity scheduling is more than sufficient**. Once you have it, click on create



Now add the necessary parameters, just like we did in the implementation section, and accept.

And that's it; you have successfully set up the automatic renewal of your SSL certificates!

Some Important Considerations

Limitations

In case you are running tests or if you have many domains/subdomains that you want to register. You must take into account the limitations established by Let's Encrypt:

- There's a limit of 20 different domains (including subdomains) per week
- There's a Duplicate Certificate limit of 5 certificates per week
- You can create a maximum of 10 accounts per IP address per 3 hours

Incompatible with Microsoft Account (Live ID)

It should be noted that it **only works with company or education accounts** (Org ID), not personal accounts (Live ID). This is because Microsoft accounts are unrelated to Azure, and the login process needs to be redirected through live.com to retrieve a token, something

impossible if done non-interactively. But don't worry, there's an easy **SOLUTION** to this problem that shouldn't take you longer than 5 minutes:

1. If you don't have an Azure AD tenant you can create one
2. Create an Azure AD user with user role
3. Go to the **resource group** of the website in which you want to set up SSL and, in the access control (IAM) section, add the user that you have created in Azure AD, and assign it a **Collaborator** user role
4. Finally, configure the Azure AD account credentials in Azure Automation, as in step 3.2.