Using the AWS Marketplace Serverless SaaS Integration Quick Start to complete your SaaS Contract product API integration

AWS Marketplace now enables sellers, Independent Software Vendors (ISVs), and Consulting Partners (CPs) to complete the AWS Marketplace SaaS product API integration in as little as 10 minutes, (excluding testing) eliminating the heavy development effort previously required. This will allow sellers to have a transactional net-new SaaS product on AWS Marketplace within hours after completing the API integration. This solution deploys AWS services as well as AWS Serverless services in your AWS account which incurs less with a pay-for-value billing model.

Until now, sellers had to invest development resources to build out their API integration with AWS Marketplace SaaS APIs and it use to take a SaaS ISV anywhere from 30-60 days to list a SaaS product on Marketplace.

Seller interested in using this solution can navigate to the AWS Marketplace Serverless SaaS Integration Quick Start.

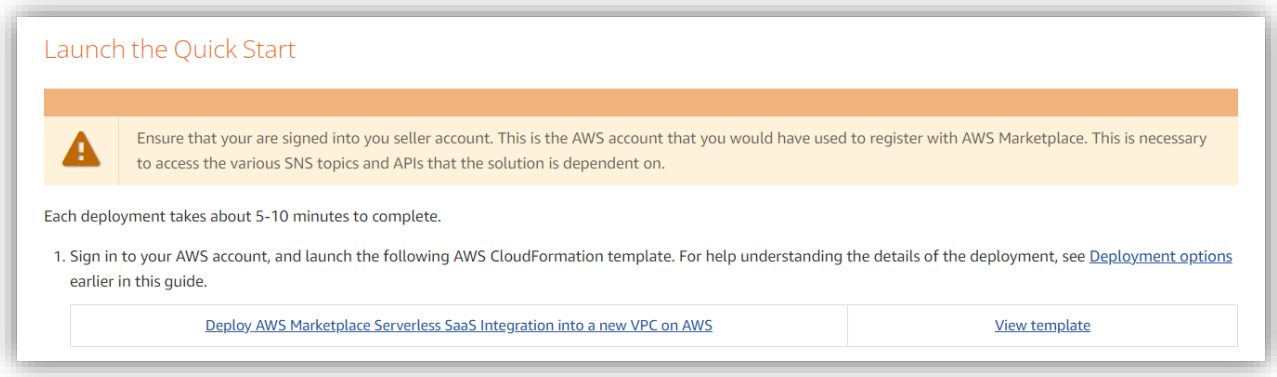
Prerequisites: Before you can begin using this solution, you must complete the following:

1. Have access to the AWS Marketplace Management Portal. This is the tool that you used to register as a seller and manage the products that you sell on AWS Marketplace.
    a. I assume you have registered as a seller in AWS Marketplace. If you have not, review the AWS Marketplace blog post, 7 Tips to Successfully Submit Your Product Listing in AWS Marketplace.
2. Create and submit a new SaaS product using the AWS Marketplace Management Portal. For more information you can review this blog, Best practices guide to successfully list your SaaS contract solution in AWS Marketplace
    a. The AWS Marketplace Catalog Operations (MCO) team will publish your product into a limited state where it is only visible to you and any AWS accounts you have whitelisted to view the product.
    b. The AWS MCO team will send an email message to the email address associated with your AWS seller account and provide the product code, Amazon Simple Notification Service (Amazon SNS) topics, and a link to your product detail page.
3. Take note of the product code, SNS topics, and link to the product detail page as you will need these as you complete the API integration.

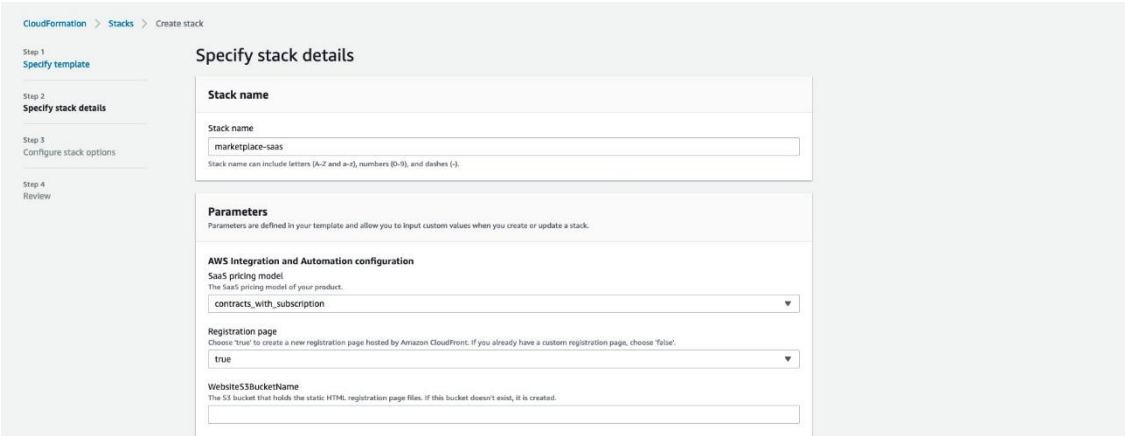
Getting started using the AWS Marketplace Serverless SaaS Integration Quick Start

1. Sign in to your AWS account that you used to register as a seller.
2. Navigate to the AWS Marketplace Serverless SaaS Integration landing page.
3. Click the **View deployment guide** button. This will launch the deployment guide and walk you through completing the API integration for your SaaS solution.
4. Read through the deployment guide up until you get to the **Deployment steps** section.

5. Click the link “Deploy AWS Marketplace Serverless SaaS Integration into a new VPC on AWS” (seen in screenshot below)
    a. If you wish to view the template first, you can do so by click the **View template** link



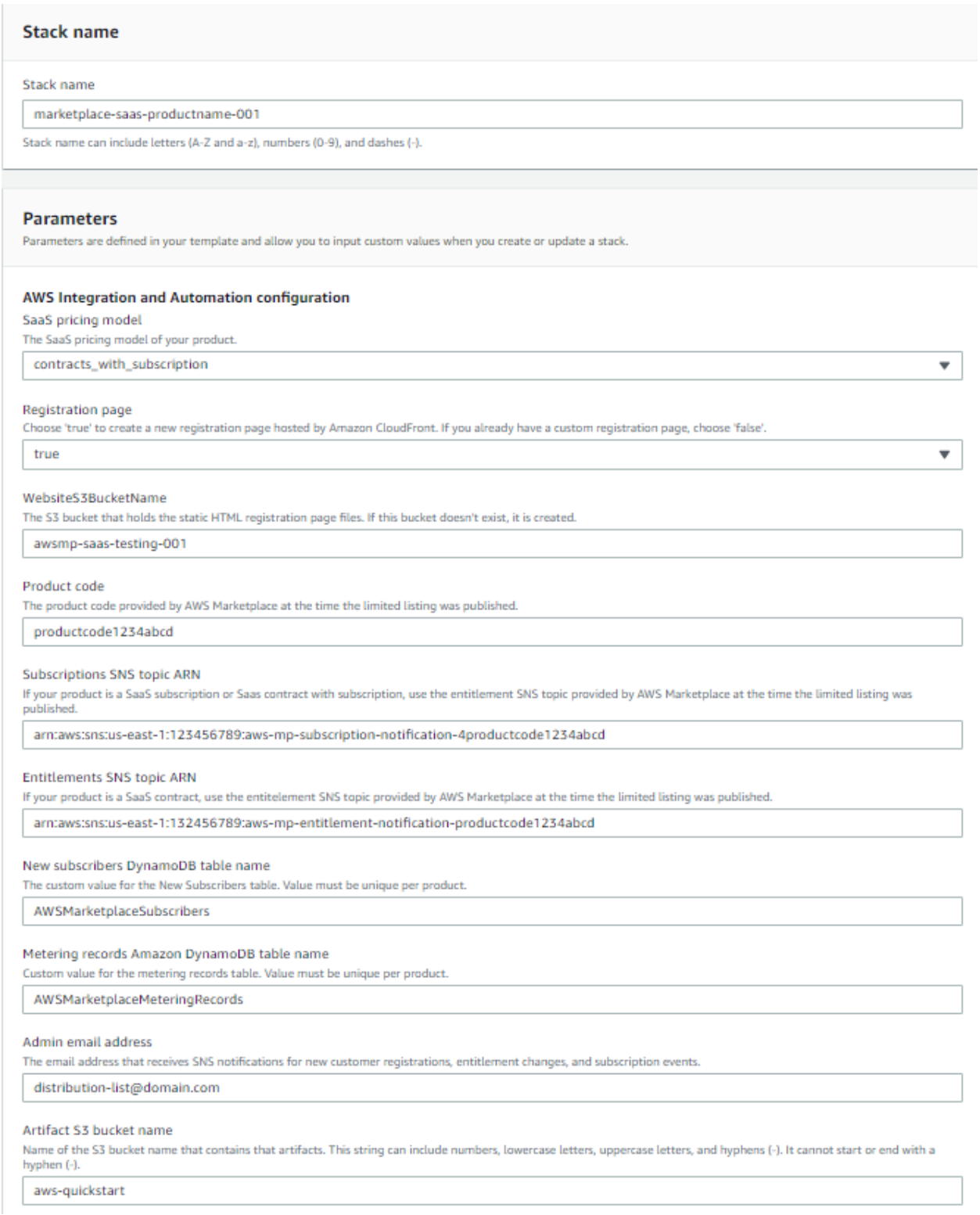
6. This will launch the AWS CloudFormation wizard to begin creating the stack
7. Ensure you are in us-east-1 (N. Virginia) AWS Region from the top right toolbar and leave the defaults as show in the screenshot below and click the **Next** button



    a. Provide a unique stack name in **Stack name** field, example; *marketplace-saas-productname-001*.
    b. Select the appropriate type of SaaS pricing model from the SaaS pricing model drop down (this relates directly to the type of SaaS product you created).
    c. If you wish to use the default Registration page created by this solution for your SaaS product then select **true** from the **Registration page** drop-down (for this Quick Start we recommend you select **true**).
    d. Provide a unique bucket name where static HTML pages for your product registration page will be saved during post deployment steps in the **WebsiteS3BucketName** field (be sure to follow S3 bucket naming rules)
    a. Enter the **Product code** value provided by the AWS MCO team in the Product code field.
    b. Enter the Entitlement and/or Subscription SNS topic ARNs provided by the AWS MCO team in the respective fields.

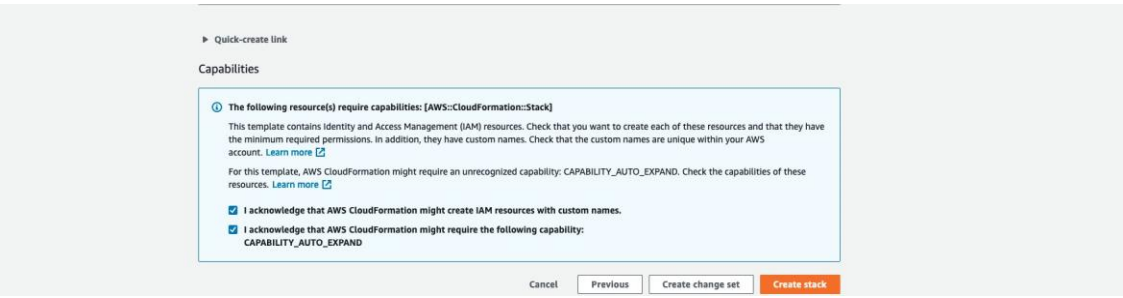
a. Enter unique names in the **New subscribers DynamoDB table name** and **Metering records Amazon DynamoDB table name** fields for storing valid customer records in Amazon DynamoDB tables (you can leave the defaults as well)
b. Provide your email address in the **Admin email address** field where emails will be sent on changes requiring action such as new subscriber notifications.
    i. **Note**: It is recommended that you use a shared email / distribution list alias for this field.
    ii. **Note**: This field cannot be modified after the Quick Start has been deployed
c. Please do not change the value for **Artifact S3 bucket name** field.

6. Review all your fields and compare to the final screenshot below
7. Click the **Next** button



7. You can keep all the options as default on the next page and then click the **Next** button

8. Review the next page and select both the checkboxes at the bottom of the review page and then click the **Create stack** button



9. Your CloudFormation stack will now deploy and should complete within 10 minutes
    a. If you run into errors please use the Contact Us link to contact the MCO team.
10. Return to the deployment guide and navigate to the Post-Deployment Steps section
    a. Download the website files to your local computer using the link in the deployment guide
    b. Unzip the file and make note of this location as you will need to update one of the files as well as copy these files to the Amazon S3 bucket you created
11. Next you will need to retrieve the API ID that was created. To do so navigate to API Gateway > Locate the **stack name** you created during deployment (example; *marketplace-saas-productname-001*) > find the ID column for your stack and copy down that ID as it is the API ID.
14. Navigate back to the file you unzipped and open the **web** folder to access the files inside
    a. Open the "script.js" file in Notepad++ or Notepad as you will need to edit the file and save it. If you have Notepad++ installed you can just right-click the "script.js" file > Edit with Notepad++
    b. Find the **baseUrl** property in line 1.
    c. Copy the API ID you just retrieved from the API Gateway console and paste that ID over the text, "API-ID" in the **baseUrl**
    d. Save the changes to the file
15. Navigate to Amazon S3 in the AWS Management console
16. Click on the S3 bucket name you provided in step 4 for **WebsiteS3BucketName field** to open the bucket. The bucket should be empty.
    a. Upload the five files from the web folder to your website S3 bucket.
17. Navigate to Amazon CloudFront in the AWS Management Console and then choose the CloudFront ID that was created by the stack in step 8. If you are unsure because you have multiple, look at the "Origins" column as the name will have your **WebsiteS3BucketName** in the URL
    a. Copy the **Domain name** as you will need to update your **SaaS URL** field for the SaaS product you created in the AWS Marketplace Management Console.
17. Navigate back to the SaaS section of the AWS Marketplace Management portal
18. Click on your SaaS product.
    a. Click the **Request changes** drop-down menu > select **Update product and pricing**
    b. Click the **General** tab > Navigate to the **SaaS URL** field

c. Paste the **domain name** you copied down earlier into **SaaS URL** field
d. Click the **Next** button > click the **Next** Button again > Enter notes such as;
    i. "We have completed the API integration using the AWS Marketplace Serverless SaaS Integration Quick Start and we have updated the SaaS URL with the new CloudFront URL. Can you please update our product and test?"

19. The AWS MCO team will publish your product with the new SaaS URL. Your product is now ready for buyer experience testing and going live on AWS Marketplace.

What's next?

1. Handling SNS Topic Notifications

During the Quick Start deployment, you entered an Admin email address that will receive SNS notifications for new customer registrations, entitlement changes, and subscription events. You will need to monitor these emails and address the notifications based on their type.

Some other best practices are below:

- Ensure the Admin email address you used during the Quick Start is a shared or distribution list and monitored on a regular basis
- Store the details from the new subscriber email in your customer database along with all the other customer information
- Ensure you reach out to new customers within 48 business hours
- To ensure a good customer experience, onboard your customers quickly
- Know when to Contact the AWS Marketplace Catalog Operations (MCO) team
    - Unsure of an email notification
    - Receiving any errors related to your API integration
    - Questions about customizing your SaaS registration page
- Document the process for your teams to follow

Types of notifications sent:

1. Customer changes contract duration.
2. Customer purchases multiple contracts when multi-buy is enabled.
3. Customer purchases higher pricing tier.
4. Customer's contract auto-renews when auto-renew is enabled.

For example, you will receive **New AWS Marketplace Subscriber** emails. Once received, it would be a best practice to save all the details contained in the email in your database along with all the other customer information.

Sample new subscriber email notification:

***Grant access to new SaaS customer**:*
{"productCode":"2p409vwjybxwn3pd5tcrz4xbw","successfully_subscribed":true,"contactEmail":"johndoe@gmail.com","created":"1645056233555","companyName":"AWSMP","subscription_expired":false,"contactPerson":"Johndoe","entitlement":"{\"Entitlements\":[{\"ProductCode\":\"2p409vwjybxwn3pd5tcrz4xbw\",\"Dimension\":\"wuphf\",\"CustomerIdentifier\":\"CbGso7gbieE\",\"Value\":{\"IntegerValue\":1},\"ExpirationDate\":\"2022-03-17T00:03:11.072Z\"}]}","customerIdentifier":"CbGso7gbieE","contactPhone":"9292685011"}

Sample notification for customer purchasing additional contracts:

***New entitlement for customer**:*
*{"productCode":"2p409vwjybxwn3pd5tcrz4xbw","successfully_subscribed":true,"contactEmail":"johndoe@gmail.com","created":"1645056233555","companyName":"AWSMP","subscription_expired":false,"contactPerson":"John Doe","entitlement":"{\"Entitlements\":[{\"ProductCode\":\"2p409vwjybxwn3pd5tcrz4xbw\",\"Dimension\":\"wuphf\",\"CustomerIdentifier\":\"CbGso7gbieE\",\"Value\":{\"**IntegerValue\":10**},\"ExpirationDate\":\"2022-03-17T00:03:11.072Z\"}]}","customerIdentifier":"CbGso7gbieE","contactPhone":"9292685011"}*

2. Marketing planning

Start to think about marketing activities to drive demand for your solution. Think of AWS Marketplace as a procurement channel for you to transact with AWS customers. It is critical that you have plans to market your solution. To do that, consider completing the following activities:

- Join the AWS Partner Network (APN), which is a program focused on providing members of the APN with programmatic, technical, business, and go-to-market support.
- If you are already part of the APN, begin to think about how you can start co-selling with AWS by participating in the APN Customer Engagement Program (ACE).
- Create a press release regarding your launch in AWS Marketplace.
- Consider authoring a blog post for the AWS Marketplace blog. These blog post should be a tutorial in nature and show how to solve a common customer challenge using your product and other AWS services.
- **AWS Marketplace 180-day GTM Academy**, an online portal that provides self-service go-to-market (GTM) resources to help you build, activate, and measure demand generation campaigns for your offerings in AWS Marketplace.
    - To register for the 180-day GTM Academy portal, log into your AWS Marketplace Management Portal (AMMP).

- If you are a first-time user, register first.
- In the Marketplace Resources or the Announcements section, choose the 180-day GTM Academy link.
- Once registered, you can sign in at 180dayGTMAcademy.com.

3. Customizing Your Registration Page

If you want to add your logo to the registration page,

- Use the *.png* version of your Logo and rename it to "logo.png"
- Navigate to Amazon S3 in the AWS Management console
- Click on the S3 bucket name you provided in step 4 for WebsiteS3BucketName field to open the bucket.
- Upload your logo file to the S3 bucket. It will replace the existing file.
- Wait 2 minutes and go to the registration page to review the new logo.

4. Getting answers

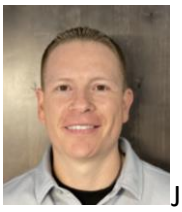
If you have questions on listing and getting started, use the following resources:

- If you are a startup, check to see if you qualify for the AWS Marketplace Startup program.
- Get registered as a seller. In the management portal, use the Contact Us form to request to speak with a member of our Marketplace Business Development team.
- Contact your AWS Account Manager and have them get in contact with the AWS Marketplace Emerging Tech Business Development Team.

Conclusion

In this post, I showed how to successfully complete the SaaS API integration required to go live in AWS Marketplace with a SaaS solution. I also touched on some of the marketing best practices you should consider. If you have additional questions contact us using the channels listed in the Questions on listing or getting started section earlier.

About the authors



Juston Salcido is a Technical Business Development Manager focusing on helping Independent Software Vendors (ISVs) understand how to list their solution in AWS Marketplace. In his role he helps ISVs take their current solution, pricing, and sales motions and provide them with the best practices to build a successful listing in AWS Marketplace. Juston has over 10 years of experience working in the technology industry and 5 years of experience working at Amazon, 4 of which have been spent working in AWS Marketplace. Juston is located

in Bozeman, MT, and enjoys golfing, playing softball, and traveling to complete in amateur billiards competitions.

 Sumeet Gujaran is a Technical Account Manager focusing on seller operations for AWS Marketplace and is passionate about blockchain, fintech and serverless computing. Outside of work he enjoys re-watching Office, trying "parkour", cooking and playing football(soccer).