Let Us Create An IoT Inside the AWS Cloud

Sudip Chakraborty [1] & P. S. Aithal [2]

[1] D.Sc. Researcher, Institute of Computer Science and Information sciences, Srinivas University, Mangalore-575 001, India,
OrcidID: 0000-0002-1088-663X; E-mail: sudip.pdf@srinivasuniversity.edu.in

[2] Vice Chancellor, Srinivas University, Mangalore, India,
OrcidID: 0000-0002-4691-8736; E-Mail: psaithal@gmail.com

Area of the Paper: Computer Science.
Type of the Paper: Experimental Research.
Type of Review: Peer Reviewed as per |C|O|P|E| guidance.
Indexed In: OpenAIRE.
DOI: https://doi.org/10.5281/zenodo.7726980
Google Scholar Citation: IJCSBE

How to Cite this Paper:
Chakraborty, S., & Aithal, P. S., (2023). Let Us Create An IoT Inside the AWS Cloud. *International Journal of Case Studies in Business, IT, and Education (IJCSBE), 7*(1), 211-219. DOI: https://doi.org/10.5281/zenodo.7726980

International Journal of Case Studies in Business, IT and Education (IJCSBE)
A Refereed International Journal of Srinivas University, India.

Crossref DOI: https://doi.org/10.47992/IJCSBE.2581.6942.0253

Paper Submission: 18/02/2023
Paper Publication: 13/03/2023

Let Us Create An IoT Inside the AWS Cloud

Sudip Chakraborty [1] & P. S. Aithal [2]

[1] D.Sc. Researcher, Institute of Computer Science and Information sciences, Srinivas University, Mangalore-575 001, India,
OrcidID: 0000-0002-1088-663X; E-mail: sudip.pdf@srinivasuniversity.edu.in

[2] ViceChancellor, Srinivas University, Mangalore, India,
OrcidID: 0000-0002-4691-8736; E-Mail: psaithal@gmail.com

ABSTRACT

Purpose: *The Internet of Things (IoT) has become essential to modern technological advancements, with various industries leveraging its capabilities. One of the significant challenges of IoT implementation is the storage, processing, and analysis of the vast amounts of data generated. AWS is a popular platform for IoT operation. This paper proposes using Amazon Web Services (AWS) as a cloud computing platform for IoT applications. By leveraging AWS's robust infrastructure, scalability, and cost-effectiveness, IoT developers can focus on creating innovative applications that transform businesses and improve customer experiences. The interface of the various AWS service has changed. Using the New AWS IoT console, we demonstrate how to create IoT device shadows.*

Design/Methodology/Approach: *IoT Thing is the backbone of any IoT interaction. First, we create Our AWS account. Because without an account, it cannot operate IoT devices. After a successful account setup, we create our things. After that, we create a device shadow, which is used to update or fetch the device data among the devices interconnected through the Internet. There are several ways to interact with device shadows. Here we talk about the few most straightforward and efficient ways, the hardware devices like ESP8266/ESP32 wifi modules, using MQTT and IoT device clients. Another efficient way is communicating through the Lambda function, which can create an Alexa-operated IoT device.*

Findings/Result: *There are several ways to create an internet of things. Probably the best and most reliable platform is AWS. Its security and interface are user-friendly. Here, through a simple illustration, we discussed how to create Things inside the AWS cloud, the backbone of transferring or fetching data from one end to another over the Internet.*

Originality/Value: *The several documents available over the web for understanding and creating the AWS IoT. Here, through simple illustration, we demonstrate how we can easily use AWS's new interface. Our researchers working or going to experiment with AWS's new IoT console can get valuable reference information for their research.*

Paper Type: *Experimental-based Research.*

Keywords: AWS IoT, AWS shadow creation, AWS IoT using New Interface.

1. INTRODUCTION :

The Internet of Things (IoT) changes how businesses operate and interact with customers. IoT is a network of interconnected physical devices and objects that collect and exchange data. With the rapid growth of IoT, organizations are looking for ways to manage and analyze the massive amounts of data generated by these devices. Amazon Web Services (AWS) is a cloud computing platform that provides a flexible and scalable infrastructure to build and deploy IoT applications. AWS offers a wide range of services that can be used to develop and deploy IoT applications in the cloud. This research paper explores the benefits of IoT creation inside the AWS cloud. The paper is organized as follows: Section 2 provides an overview of related work already done on IoT. Section 3 discusses the objective of the research work. Section 4 highlights the methodology we used for the research work. In section 5, we do the actual experiment. This section describes the procedure with a required screenshot to create an IoT in the AWS cloud. Section 6 provides recommendations for reading or watching videos to

understand the research topic better. Finally, Section 7 concludes the paper and provides future research directions.

2. RELATED WORKS :

Al-Turjman et al. in paper propose a framework for building IoT applications on AWS. The framework includes data collection, storage, processing, and visualization components. The author argues that AWS provides a robust platform for building IoT applications and that the framework can help developers create applications more efficiently [1]. Chen, Z. describes the design and implementation of an IoT system based on the AWS cloud platform. The system includes devices for data collection, an AWS IoT Core for data processing, and an AWS DynamoDB for data storage. The authors argue that using AWS can improve scalability and reduce costs compared to traditional on-premise solutions [2]. Fujita, H. presents the development of an IoT system using AWS IoT and AWS Lambda. The system includes sensors for data collection, an AWS IoT Core for data processing, and AWS Lambda for data analysis. The authors argue that using AWS can simplify the development process and make it easier to scale the system [3]. Han, X., describes developing an IoT-based monitoring system for a greenhouse environment in the AWS cloud. The system includes sensors for data collection, an AWS IoT Core for data processing, and an AWS DynamoDB for data storage. The authors argue that the system can help improve crop yields and reduce environmental impact [4].

3. OBJECTIVES :

The Internet of things is the backbone of device control over remote. Researchers sometimes use IoT to control their experimental devices from remote places or fetch device parameters with a specific interval for their research work. In the IoT field, AWS is one of the dominant players. In the AWS IoT, device shadow is one of the IoT control mechanisms. Using the device shadow, we can do lots of things. This paper provides practical information to create an AWS IoT interface. Recently, the AWS configuration interface has changed. Our approach is to provide guidelines to engage less time to sync with the new interface.

4. APPROACH AND METHODOLOGY :

Figure 1 depicts the interaction methodology with AWS cloud device shadow. The figure shows four well-known ways to interact with AWS device shadows. Every way must follow some software configuration, parameters, and setup to communicate with AWS cloud. For Hardware like ESP8266/ESP32 and C# MqttClient, we need certificates available from AWS. When we create the things, AWS provides the keys and certificate. We need IAM user credentials for Amazon IoT DataClient and the C# lambda function. IAM user creation is tricky for the new interface. Now we will see how to create an IAM user and provide IoT access permission to the user.

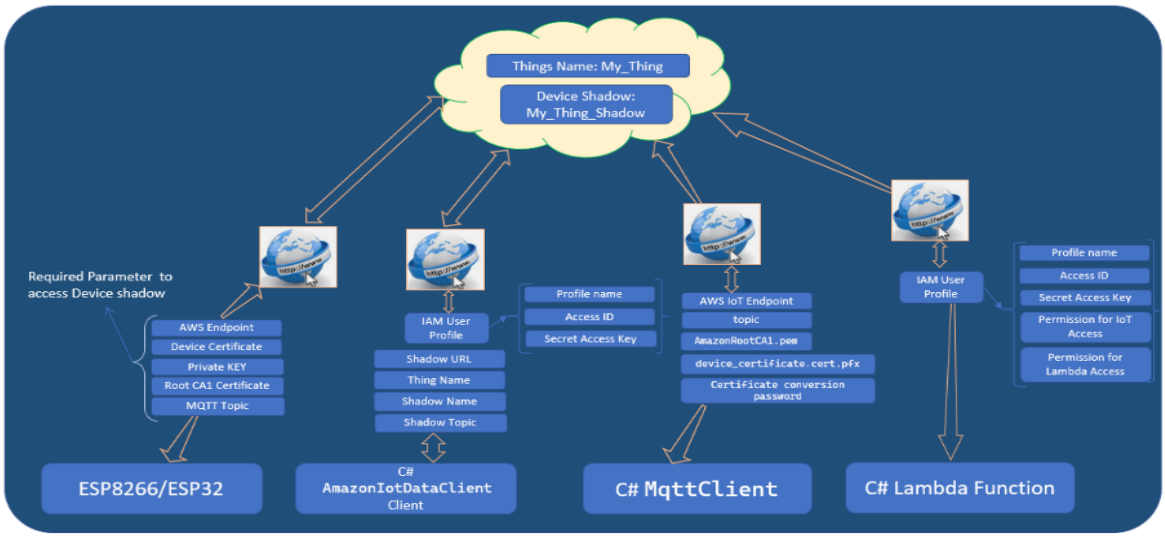

Fig. 1: Way to interact with AWS IoT device shadow methodology

Create IAM User:

1. Go to the AWS console. Search for IAM. Click the "**Users**" menu.
2. Click on the "**Add users**" Button at the top right side.
3. Inside the user name textbox, type "**My_Thing_User**." Click on the "**Next**" Button.
4. Keep "**Add user to group**" selected and click the "**Next**" Button. Click on the "**Create user**" Button.
5. Now the just-created user is displayed on the page.

Add Permission to the IAM User:

1. Click on "**My_Thing_User**."
2. Select the "**Permissions**" tab.
3. On the right side of the page, click on the "**Add permissions**" Button. From the drop-down list, select "**Add permissions**."
4. Select the "**Attach policies directly**" radio button. An available permissions list will be displayed.
5. Inside the Find policies search box, type "AWSIoTfullAccess."
6. Only one option will be displayed. Select the checkbox and click on "**Next**." Click on "**Add permissions**."
7. Figure 2 depicts after permission was added.

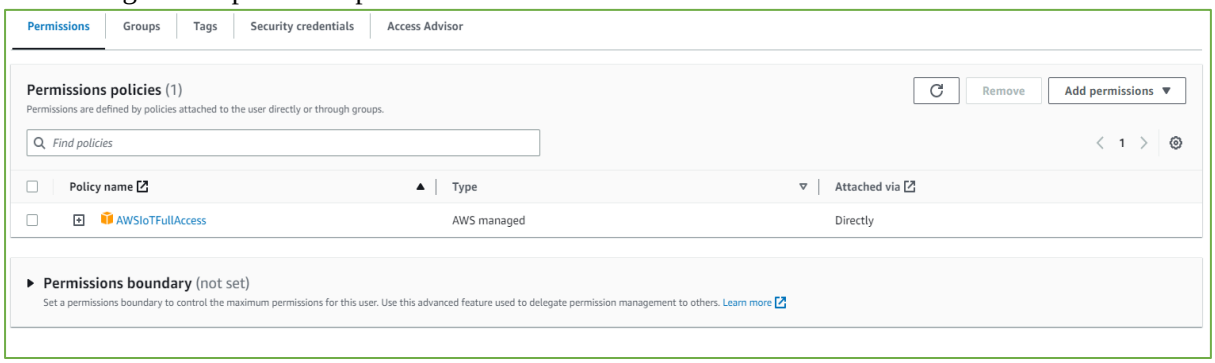

Fig. 2: permission added to the IAM user

Generate access key and secret key for IAM user:

We must create Access and secret keys to access the device shadow from our PC.

1. From the Left, click on the user. Click on "My_Thing_User." Select the "**Security credentials**" tab.
2. Scroll down. Under the "Access keys" section" click on the "**Create access key**" Button.
3. Select "**Application running outside AWS**" and click "**Next**."
4. Under the Description tag value, write "for IoT Access from PC."
5. Click on the "**Create access key**."
6. Copy the Access and Secret access keys and keep them safe. The option is also available to download the .csv file.
7. The Access key looks like "AKIAUA2[redacted]UYT7ZY," and Secret access key looks like "+Bkr0dKUqS90mm6h8C[redacted]j7F2riP+aEUs1TfM"
8. Click on the "**Done**" Button.
9. Close the browser.
10. Figure 3 depicts the Access key page.

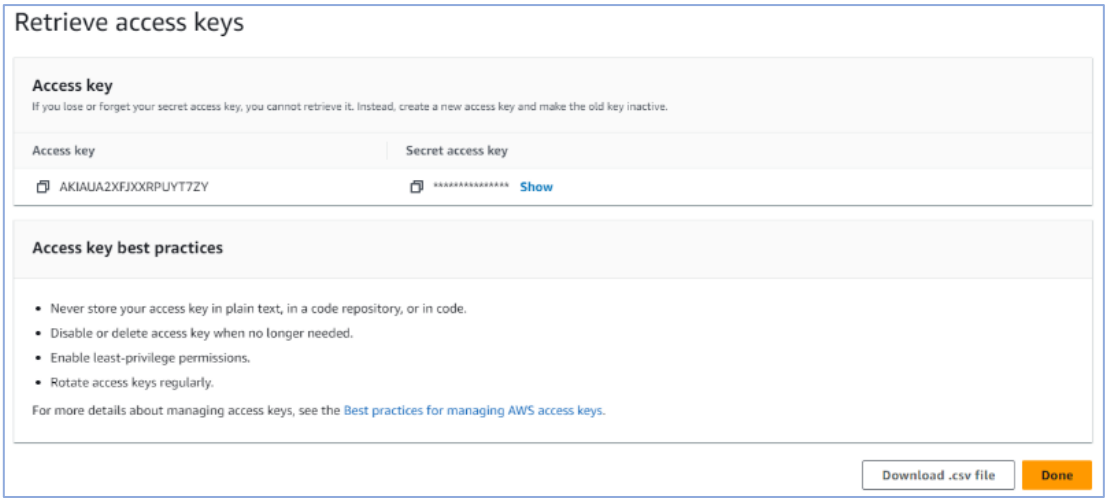

Fig. 3: Access key retrieval

When we access AWS cloud services like IoT from our PC or laptop, outside of AWS cloud, we need to Add an IAM profile to the Visual Studio Environment. Before this execution, we must install AWS SDK. NET.

1. Open visual studio.
2. From the view menu, open Aws explorer. On the top portion of the window, click on the "+" Button("Add AWS credentials Profile" button).
3. Add the profile name.
4. Paste the "Access key ID" and "Secret Access Key."
5. Select the region and press OK.
6. Figure 4 depicts the credentials entry page.

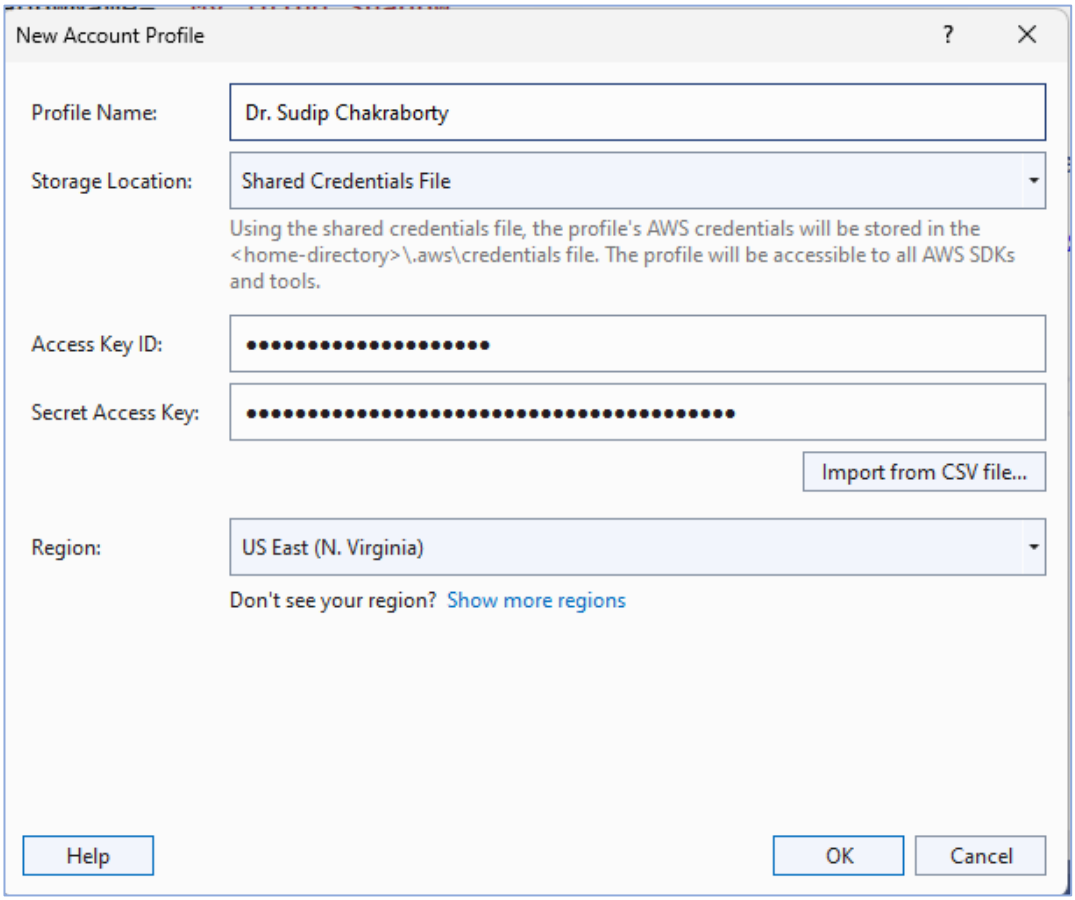

Fig. 4: Visual Studio Credential Entry Page

5. EXPERIMENT :

Let us start our experiment. Through the below steps, we will create things in the AWS cloud.

Step 1: Create an AWS IoT account

1. To do IoT-related activities, we need to create an AWS account. Search in google "AWS IoT console." Open the link.
2. Keep the radio button "**Root user**" selected. At the page's bottom, click "**Create a new AWS account**". The signup page will appear.
3. Select the "**Personal for your projects**" radio button. Then fill Full name, country, address, city, state, and postal code.
4. Checked on the "I have read and agree to the terms of the AWS Customer Agreement" check box. click on "**Continue(step 2 of 5)**".
5. The billing Information will appear. Fill in the Credit or Debit card number, Expiration date, month, year, Card holder name, CVV, and yes/no for "Do you have a PAN?". click on "**Verify and Continue(step 3 of 5)**."
6. The "Confirm your identity" page will appear. Select "Text message (SMS)." Select Country or region code. Enter your mobile number. Type the Security check code inside the textbox. Click on "**Send SMS(step 4 of 5)**".
7. Check your mobile phone. Enter the verification code inside the textbox received on the mobile phone. Then click on "**Continue(step 4 of 5)**".
8. The "Select a support plan" page will appear. Select "Basic support-Free" and click on "**Complete signup**".
9. Now the "**Personalize your Experience**" page appears. Select My role is" **Academic/Researcher**." Next, I am interested in: the "**Internet of Things**" and clicking the "Submit" Button.

Step 2: Introduce AWS cloud Interface

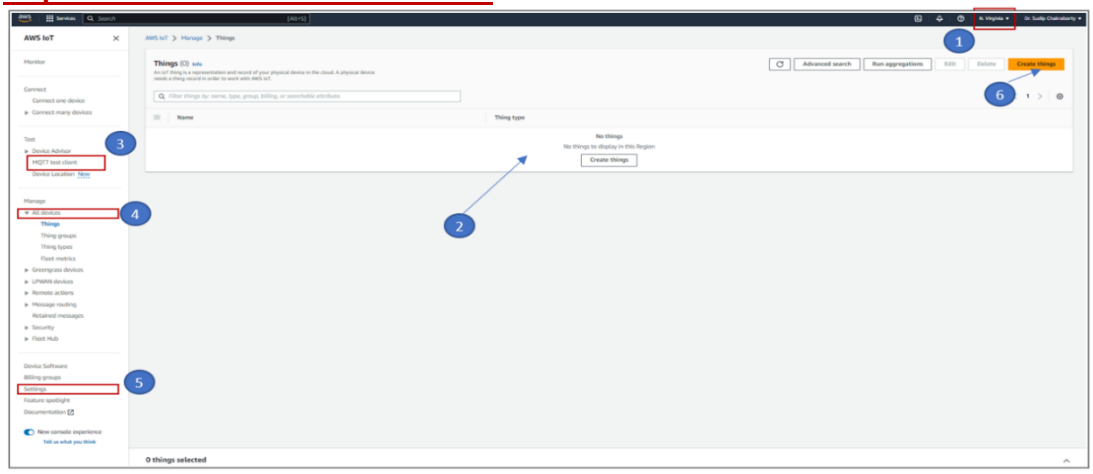

Fig. 5: AWS cloud menu structure

Figure 5 depicts the position of the GUI control item.

(1) This is one of the items to take care of. This is the region server where we create our Things. Before we start to create the IoT, we need to check carefully. The nearest server region takes less time to execute the operation—selecting Improper regions cannot fetch the data.
(2) After thing creation will be displayed here.
(3) This menu is responsible for testing MQTT publish or subscribe.
(4) This Left side menu is the entry point for any operation on things. It also helps to open our created Things.
(5) We get the Device Endpoint URL and device configuration-related operation from this menu.
(6) This Button is used to create new things and device shadows.

Step 3: Create AWS IoT things

1. Create an IoT account in AWS. Open AWS Console > Search IoT Core and Open.
2. Select zone. Like US East(N. Virginia), i.e., us-east-1.
3. Open IoT Dashboard. from the left side, Click manage.
4. Navigate All Devices>Things>click on the "**Create things**" Button>select "**Create single thing**." click on "**Next**."
5. Thing name: "**My_Thing."**
6. Device Shadow: "No shadow."
7. Click **Next**.
8. Under the Device certificate page, keep selecting "Auto-generate a new certificate(recommended)" >click **Next**> click "**create things**."
9. Under the "Download certificates and keys" page, download four certificates.
    a. Click on the download button of the Device certificate. It looks like "*5dc3d7a[redacted]4...te.pem.crt*".
    b. Download the public key file. It looks like "*5dc3d7a7[redacted]fb5349d09b...1a007a2-public.pem. the key*".
    c. Download the Private key file. It looks like "*5dc3d7a7bc4[redacted]5349d09b...a007a2-private.pem. the key*".
    d. Download Root CA certificates. It looks like *"RSA 2048 bit key: Amazon Root CA 1".*
10. Click the "**Done**" Button.

Step 4: Create Policies.

1. From the Left side, under security, click the "Policies" menu.
2. Click on the "**Create policy**" button> Name: "**My_Thing_Policy**."
3. At the lower portion, under the Policy document section, "**Policy effect,**" select "**Allow**."
4. The next Item is "**Policy action**"; select "*."
5. The next Item is "**Policy resource**," type "*****." Then click Create.
6. Figure 6 depicts the policy creation interface.

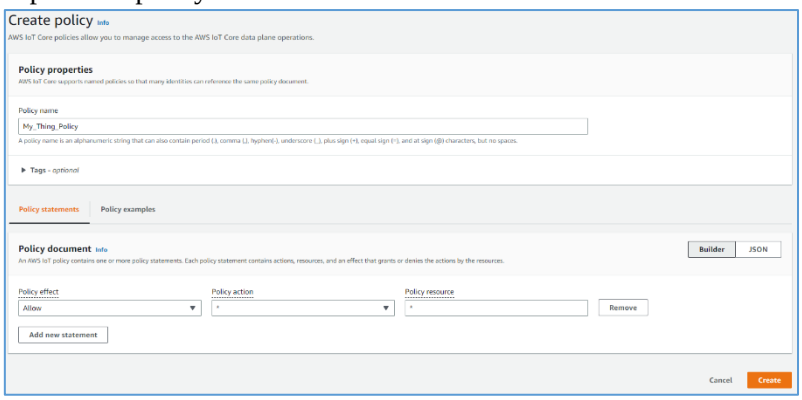

Fig. 6: Create policy Interface.

Step 5: Attach Policies with Certificate:

1) Navigate Manage>All devices>Things> click on things name "**My_Thing"**
2) Under the Certificates tab, click on Certificate ID. It looks like "*5dc3d7a7bc41ffb5[redacted]b3744fcb68b67 e5854d0379152b7ceef0731a007a2.*"
3) On the bottom-right side of the page, Click on the "**Attach policies**" Button.
4) From the combo box, select policy name "My_Thing_Policy" > click on "**Attach policies**."
5) Figure 7 depicts the attached policy interface.

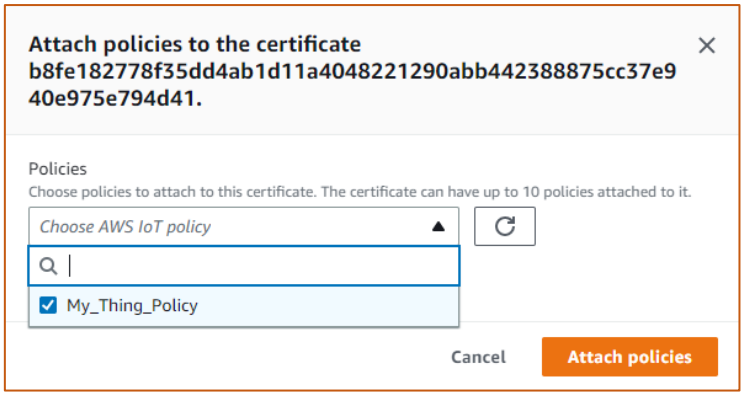

Fig. 7: Attach policy Interface.

Step 6: Create Shadow:

1) Under the Things menu, click on Things Name "**My_Thing**." select the tab "**Device Shadows**."
2) Click on "**Create Shadow**" Keep the radio button "**Named Shadow**" selected.
3) In the "Device Shadow name" textbox, type "**My_Thing_Shadow**" and click **create**.
4) Figure 8 depicts the shadow creation interface.

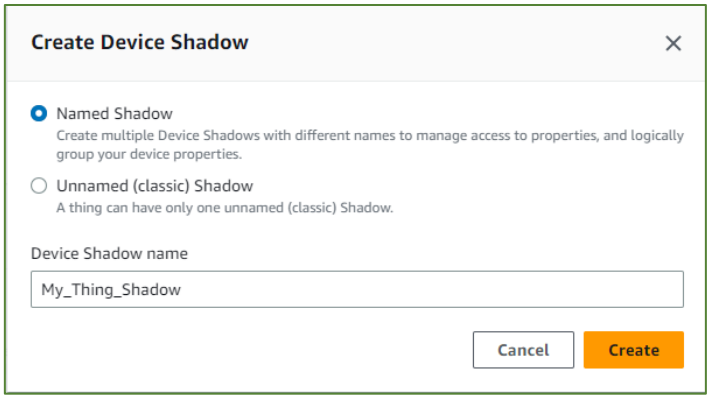

Fig. 8: Shadow creation interface

6. RECOMMENDATIONS :

- Download AWS SDK for dot net developer: https://aws.amazon.com/sdk-for-net/
- This proposed research document is the basics of the IoT for AWS. Using this Device shadow, we can connect, update and fetch data from a remote node or endpoint.
- This device shadow is used to trigger the device using Alexa voice service. Some good videos are available on Youtube to continue the research work.
- After this project execution, we can try to connect with device shadow using c# dot net. A good tutorial video: https://www.youtube.com/watch?v=ryUnPzPO44E&t=15s
- A good tutorial video on AWS IoT using Esp32: https://www.youtube.com/watch?v=ryUnPzPO44E&t=15s

7. CONCLUSION :

In conclusion, the Internet of Things (IoT) has revolutionized how we interact with technology and has brought forth endless possibilities for innovation. Integrating IoT with the Amazon Web Services (AWS) cloud platform provides numerous advantages, such as scalability, flexibility, and accessibility. This paper has demonstrated the benefits of utilizing AWS IoT services to create a robust and secure IoT infrastructure. AWS IoT offers a suite of services that can be tailored to meet the specific needs of an IoT project. These services include device management, data processing and analytics, and secure

communication. The integration of these services allows for the creation of an end-to-end IoT solution that can be scaled to meet the demands of any project.

REFERENCES :

[1] Aleem, A. and Ryan Sprott, C. (2013). Let me in the cloud: analysis of the benefit and risk assessment of cloud platform. *Journal of Financial Crime, 20*(1), 06-24. Google Scholar↗

[2] D. Gannon, R. Barga, and N. Sundaresan, (2017). Cloud-Native Applications. *IEEE Cloud Computing, 4*(5), 16-21. Google Scholar↗

[3] Tawalbeh, Lo'ai, Fadi Muheidat, Mais Tawalbeh, and Muhannad Quwaider (2020). IoT Privacy and Security: *Challenges and Solutions. Applied Sciences, 10*(12), 4102, 1-7. Google Scholar↗

[4] Rehr, J. J., Vila, F.D., Gardner, J. P., Svec, L. and Prange, M. (2010). Scientific Computing in the Cloud. *Computing in Science & Engineering, 12*(3), 34-43. Google Scholar↗

[5] Lin, G. Fu, D., Zhu, J. and Dasmalchi, G. (2009). Cloud Computing: IT as a Service. *IT Professional, 11*(2), 10-13. Google Scholar↗

[6] Nabil Sultan (2010). Cloud computing for education: A new dawn? *International Journal of Information Management, 30*(2), 109-116. Google Scholar↗

[7] Schad, J., Dittrich, J., & Quiané-Ruiz, J. A. (2010). Runtime measurements in the cloud: observing, analyzing, and reducing variance. *Proceedings of the VLDB Endowment, 3*(1-2), 460-471. Google Scholar↗

[8] Garg, S. K., Versteeg, S., & Buyya, R. (2013). A framework for ranking of cloud computing services. *Future Generation Computer Systems, 29*(4), 1012-1023. Google Scholar↗

[9] Grosse, E., Howie, J., Ransome, J., Reavis, J., & Schmidt, S. (2010). Cloud computing roundtable. *IEEE Security & Privacy, 8*(6), 17-23. Google Scholar↗

[10] Gulabani, S. (2018). Amazon Web Services Bootcamp: Develop a scalable, reliable, and highly available cloud environment with AWS. Packt Publishing Ltd. Google Scholar↗

****