**Lab 3: Using AWS S3 to Store Selenium Screenshots**

Objective:*Integrate AWS services with Selenium tasks.*

Tasks:

1. Create an S3 bucket.

2. Modify the Selenium script to upload screenshots to the S3 bucket.

Documentation:

- Introduction to AWS S3.

- Integrating Python Boto3 with Selenium.

Prerequisites:

1- An AWS account with administrative access.

2- Python Automation Course

3- Python Selenium Course

4- Bash Script Deep Dive Course

5- Previous Lab completed

Implementation Documentation:

**Introduction to AWS S3**

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. It is designed to store and retrieve any amount of data from anywhere on the web.

**Step 1: Creating an S3 Bucket**

Log in to your AWS account.

Go to the S3 dashboard.

Click on "Create Bucket".

Enter a unique bucket name and select a region.

Configure optional settings (leave default if unsure).

Set permissions (leave default if unsure).

Review and create the bucket.

**Integrating Python Boto3 with Selenium**

Boto3 is the Amazon Web Services (AWS) SDK for Python, which allows Python developers to write software that uses services like Amazon S3 and Amazon EC2. We'll use Boto3 to interact with the S3 bucket from our Selenium script.

**Step 3: Creating Access Keys for AWS**

Open IAM

Select Your User

Goto Security Credentials tab

Scroll down and click create access keys

Select the SSH option

Click create

Download the csv file and keep it safe

**Step 3: Modifying the Selenium Script**

Make sure to install boto3 in Pycharm for this project

Create a new python file in PyCharm

| # Import necessary libraries from selenium import webdriver import boto3  # Initialize the S3 client s3 = boto3.client('s3',  aws\_access\_key\_id='YOUR\_ACCESS\_KEY\_ID',  aws\_secret\_access\_key='YOUR\_SECRET\_ACCESS\_KEY',  region\_name='YOUR\_REGION')  # Initialize the WebDriver (e.g., Chrome) driver = webdriver.Chrome()  # Open the AWS website driver.get("https://aws.amazon.com/")  # Capture a screenshot driver.save\_screenshot("screenshot.png")  # Upload the screenshot to the S3 bucket bucket\_name = 'sel-test-bucket' object\_name = 'screenshot.png' s3.upload\_file('screenshot.png', bucket\_name, object\_name)  time.sleep(10) driver.quit() |
| --- |

**Explanation:**

We import the necessary libraries, including selenium for web automation and boto3 for AWS interaction.

We initialize the S3 client using your AWS credentials and region.

The Selenium script remains the same as in Lab 2.

After capturing the screenshot, we upload it to the S3 bucket using s3.upload\_file().

**Conclusion:**

By following these steps, you have integrated AWS S3 with your Selenium tasks. Now, your screenshots will be automatically uploaded to the S3 bucket. This lab demonstrates the power of combining Selenium automation with AWS services.