AWS-6

AWS CLI

1

Introduction

- So far, we've interacts with services manually and they exposed standard information for clients:
  - EC2 exposes a standard Linux machine we can use any way we want
  - RDS exposes a standard database we can connect to using a URL
  - ASG / ELB are automated and we don't have to program against them
  - Route53 and S3 was setup manual

- Developing against AWS has two components:
  - How to perform interactions with AWS without using the Online Console?
  - How to interact with AWS Proprietary services? (S3, DynamoDB, etc…)

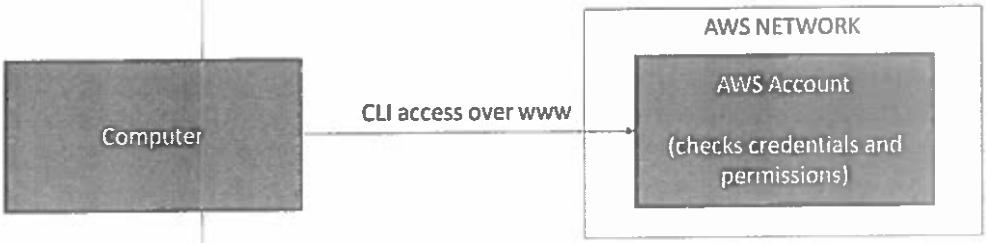
2

Introduction

- Developing and performing AWS tasks against AWS can be done in several ways
  - Using the AWS CLI on our local computer
  - Using the AWS CLI on our EC2 machines
  - Using the AWS Instance Metadata Service for EC2

3

AWS CLI Setup on Linux



- We'll learn how to get our access credentials and protect them
- Do not share your AWS Access Key and Secret key with anyone!

4

AWS CLI ON EC2... THE BAD WAY

- We could run `aws configure` on EC2 just like we did (and it'll work)
- But... it's SUPER INSECURE
- NEVER EVER EVER PUT YOUR PERSONAL CREDENTIALS ON AN EC2
- Your PERSONAL credentials are PERSONAL and only belong on your PERSONAL computer

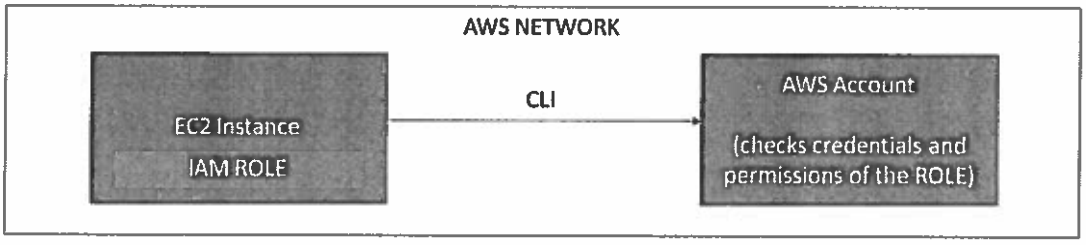
- If the EC2 is compromised, so is your personal account
- If the EC2 is shared, other people may perform AWS actions while impersonating you

- For EC2, there's a better way... it's called AWS IAM Roles

5

AWS CLI ON EC2... THE RIGHT WAY

- IAM Roles can be attached to EC2 instances
- IAM Roles can come with a policy authorizing exactly what the EC2 instance should be able to do



- EC2 Instances can then use these profiles automatically without any additional configurations
- This is the best practice on AWS and you should 100% do this.

6