Part 1: Identify the Problem

Unused AWS Resources and Their Impact

Idle AWS resources, such as EC2 instances, unattached EBS volumes, inactive S3 buckets, unused Elastic IPs, and dormant IAM users or roles, lead to increased costs, security vulnerabilities, and added complexity in resource management. Automating their cleanup mitigates these issues by optimizing resource utilization, reducing costs, and enhancing security.

Relevance to Cloud Computing

This solution is integral to cloud infrastructure management and automation. By leveraging the boto3 library, it showcases efficient AWS service management and aligns with cloud operations' best practices. The project demonstrates expertise in automating repetitive tasks, improving cost efficiency, and strengthening security within cloud environments.

Approach

1. Libraries and Tools

* boto3: For programmatic interaction with AWS services.
* InquirerPy: For interactive command-line prompts, enabling user inputs during execution.
* logging: To track actions and record activity logs for debugging and auditing.

2. Workflow

Step 1: Authentication and Setup

* Use AWS credentials configured through the AWS CLI or a credentials file.
* Ensure appropriate IAM permissions are available for resource cleanup operations.

Step 2: Resource Identification

* Query AWS services using boto3 to identify unused or idle resources:
  + EC2 instances: Detect instances running beyond a specified threshold.
  + EBS volumes: Identify unattached volumes.
  + S3 buckets: Find buckets inactive for over 30 days based on access logs.
  + Elastic IPs: Locate IPs not associated with any instance.
  + IAM users/roles: Detect accounts or roles unused for 90+ days.

Step 3: Interactive Cleanup

* Use InquirerPy to prompt users for confirmation before performing deletion tasks.
* Present a list of detected resources for review and manual selection.

Step 4: Error Handling

* Gracefully handle missing permissions, API rate limits, and other exceptions (e.g., AccessDenied, Throttling, or ResourceNotFound).
* Safeguard critical resources tagged with labels like "DO NOT DELETE."

Step 5: Logging and Reporting

* Record all actions and detected resources in log files.
* Provide a summary of operations upon script completion.