Incident response with AWS detective controls

Dmytrii Tykholaz[1,†], Roman Banakh[1,†], Lesya Mychuda[1,†], Andrian Piskozub[1,†] and Roman Kyrychok[2,*,†]

[1] *Lviv Polytechnic National University, 12 Stepana Bandery str., 79013 Lviv, Ukraine*
[2] *Borys Grinchenko Kyiv Metropolitan University, 18/2 Bulvarno-Kudryavska str., 04053 Kyiv, Ukraine*

Abstract

Loss of access to an Amazon Web Services (AWS) account can be caused by a variety of factors, ranging from human errors to hacks. Such incidents can have serious consequences for both companies and individuals, especially in the context of safeguarding critical data and ensuring business continuity. One of the most effective strategies for preventing AWS account loss is automating security configurations and incident response processes. Utilizing AWS Detective Controls tools in combination with deployment through Terraform addresses these challenges, enhancing security and resilience to potential attacks. The combined use of AWS Detective Controls and Terraform enables organizations to achieve enhanced security and efficiently respond to incidents. AWS Detective Controls acts as a constant monitoring environment, detecting potential threats and giving administrators the ability to quickly react to any incidents that could cause account loss. Terraform automates security configuration processes across the entire infrastructure, ensuring unified security rules for all AWS services. With such solutions, organizations can control their AWS accounts at a more advanced level, reducing the risk of account loss and ensuring protection against cyber threats. This is especially important in today's world, where cloud services play a key role in business processes and data protection. This topic is particularly relevant because securing information systems in cloud environments such as AWS presents a significant challenge for many companies. Research into investigating incidents using AWS Detective Controls helps improve awareness and establish new or improved approaches to detecting various security threats and swiftly responding to breaches. The results of such research will be useful for any organization using AWS, as well as for cybersecurity researchers, especially those working to improve the protection of cloud infrastructures.

Keywords

information security, cybersecurity, AWS, account loss, cybersecurity, data protection, incident response, AWS Detective Controls, Terraform, security automation, threat detection, security-as-code, cloud service provide, DevSecOps, cloud environment, breaches

1. Introduction

In developing this research, we built on both previous studies and our own earlier work [1] in protecting cloud infrastructure and access to it. In the research paper [2] authors proposed a framework for enterprise cloud infrastructure, while researchers in [3] focused on security challenges using a "Security-as-Code" approach. In research [4] author discusses the impact of decoys involving blockchain technologies on the state of information security of the organization and the process of researching cybercrime which is very important in cloud computing environments as after gaining the access to account attackers may launch computing resources for their benefit. Often, to gain access to an account, attackers choose not to attack the system directly but to gain access through the weakest link—a person. In such cases, detecting the attack is much more challenging since access to the resource often looks legitimate [5].

Leveraging insights from these studies, as well as our prior research, we extended their methodologies to address specific gaps in cloud infrastructure deployment and security management, proposing solutions aimed at enhancing security resilience by protecting cloud resources and reducing the amount of misconfiguration that might occur due to manual mistakes [6].

For 2024 stolen credentials are the leading cause of data breaches—16% of all incidents with data compromised, contributing to a significant increase in intellectual property theft, following the IBM Report: Escalating Data Breach Disruption Pushes Costs to New Highs [7]. These breaches take an average of 10 months to detect and contain, resulting in substantial financial losses.

From 2023 to 2024, several major data breaches occurred, including:

- Critical MOVEit vulnerability led to the compromise of more than 2,300 organizations, including Shell, British Airways, the US Department of Energy, and

CPITS-II 2024: Workshop on Cybersecurity Providing in Information and Telecommunication Systems II, October 26, 2024, Kyiv, Ukraine
* Corresponding author.
† These authors contributed equally.
dmytrii.tykholaz.mkbbi.2023@lpnu.ua (D. Tykholaz);
roman.i.banakh@lpnu.ua (R. Banakh);
lesia.z.mychuda@lpnu.ua (L. Mychuda);
andrian.z.piskozub@lpnu.ua (A. Piskozub);
r.kyrychok@kubg.edu.ua (R. Kyrychok)

0000-0003-1014-5601 (D. Tykholaz);
0000-0001-6897-8206 (R. Banakh);
0000-0001-8266-1782 (L. Mychuda);
0000-0002-3582-2835 (A. Piskozub);
0000-0002-9919-9691 (R. Kyrychok)

Ontario's government birth registry, BORN Ontario, the latter of which led to the compromise of information for 3.4 million people [8].

- Boeing incident was assessing a claim made by the Lockbit cybercrime gang that it had "a tremendous amount" of sensitive data stolen from them [9].
- AT&T data-specific fields were contained in a data set released in the dark [10].
- Bank of America data breaches: a data breach exposing the personally identifiable information (PII) of 57,028 customers [11].

Data leaks often result from password theft, which remains one of the most common causes of sensitive data loss [12]. A notable example is the data leak incident involving stolen passwords that hackers used to access user accounts. For example, Boeing's stolen data was sold on the dark web, giving other cybercriminals access to private information and confidential stuff [13, 14].

These cases demonstrate that losing access to social media accounts typically results in privacy violations or loss of control over personal information. However, losing access to cloud computing accounts such as AWS can have much more severe consequences. Attackers gaining access to these accounts can not only steal critical data but also alter infrastructure settings, potentially leading to substantial financial losses, business damage, or even the shutdown of critical operations [15].

While password loss can have serious consequences, cloud computing accounts like AWS are particularly vulnerable because they typically grant access to large volumes of data and critical resources. This underscores the importance of not only creating strong passwords but also implementing multi-factor authentication and access control systems to protect such accounts.

1.1. Potential threats of AWS account loss

Amazon Web Services (AWS) is one of the leading cloud computing platforms today—thousands of companies use AWS daily to store and process their data. However, there is a risk of losing access to an AWS account, which could lead to data breaches and severe security violations. Given that AWS is a highly powerful and flexible platform offering a wide range of services and infrastructure for data storage, processing, and deployment of applications online (including more than 200 services), the likelihood of such incidents is quite significant.

Various attack vectors can lead to the loss of an AWS account. Potential threats include:

- Insufficient authentication and authorization.
- Use of weak passwords or poor password management.
- Phishing and spear-phishing attacks.
- Use of unsecured APIs.
- Inadequate network security.
- Weak authentication and authorization mechanisms.

1.1.1. Insufficient authentication and authorization

Authentication and authorization are critical security aspects in AWS. Insufficient authentication can allow an attacker to gain access to an account and take malicious actions. Insufficient authorization enables users to perform unauthorized actions, potentially leading to severe consequences.

To prevent this, AWS's best practices recommend using multi-factor authentication (MFA) [16]. This requires not only a password but also a code generated on a user's device. Additionally, AWS offers the use of authorization policies to limit users' access to resources and services. One of the key services for managing access is Identity and Access Management (IAM) [17]. AWS IAM allows for managing individual users, user groups, and access rights to various AWS account resources.

1.1.2. Use of weak passwords or poor password management

Using weak passwords or improperly storing passwords is also a serious security threat to AWS. If a password lacks sufficient complexity or is stored in an insecure location, an attacker can easily steal it and gain access to an account.

To prevent this, AWS recommends using complex passwords containing various symbols, numbers, and lowercase letters. Furthermore, AWS advises using different passwords for each service and storing them in a secure location, such as a password manager.

1.1.3. Phishing attacks

Phishing and spear-phishing attacks are social engineering techniques used to gain access to accounts. In the context of AWS, attackers send emails or messages that appear to be legitimate communications from AWS or another related service. These messages may contain embedded links to fake websites or malicious software that collects user information.

To avoid such situations, AWS advises users to be cautious with emails and messages from unknown or unverified sources. Additionally, AWS recommends using protection technologies that can detect and block malicious software and websites, such as antivirus and anti-phishing tools.

Inadequate data protection can also lead to the loss of an AWS account. Attackers may gain access to user data through various methods, including password and identity theft. Specifically, attackers may use unsecured networks and exploit software vulnerabilities to access data [18-20].

To prevent this, AWS's best practices recommend using methods such as encryption, data loss prevention via backups, and monitoring data access. Additionally, AWS recommends using secure networks and reliable communication channels for data transmission [21].

1.1.4. AWS denial of service

AWS may deny service to users who violate service conditions or cause problems for other users. For instance, AWS may suspend accounts involved in criminal activity or spamming. Additionally, if a company or individual does not pay for services, AWS may block the account.

To prevent service denial, AWS encourages users to adhere to service terms and avoid violating platform rules. Additionally, AWS recommends regular payment of bills to prevent account suspension.

1.1.5. Weak authentication and authorization mechanisms

In some cases, AWS users do not adequately support the security of their accounts. For example, users may fail to regularly update software or improperly configure security

policies, which could lead to system security breaches and account loss.
AWS encourages users to regularly update software and properly configure their security policies. Additionally, AWS recommends using various security tools, such as security monitoring, antivirus software, and data loss prevention.

2. Use of AWS detective controls for incident analysis

Given the information in the previous section, it can be concluded that the potential loss of access to an Amazon Web Services (AWS) account can have serious consequences for users, including loss of critical data, significant financial recovery costs, and severe reputational damage for businesses. To avoid such problems, users should follow cybersecurity best practices and utilize tools designed to protect AWS accounts.

Some of the most important tools that ensure security in AWS are detective controls [22], such as CloudTrail [23], VPC Flow Logs [24], GuardDuty [25], and Trusted Advisor [26]. Each of these tools plays a crucial role in maintaining system security.

- CloudTrail allows users to track actions within their accounts and log them. This helps identify suspicious or unauthorized activities, which is a critical component for preventing potential threats.
- VPC Flow Logs monitor network activity, helping detect anomalies that may indicate security threats.
- GuardDuty automatically identifies potential threats and sends security incident alerts, allowing administrators to quickly respond to threats.
- Trusted Advisor helps improve security, optimize AWS service usage, and provides recommendations for enhancing security practices.

According to the AWS "Security Incident Response Guide" [19], organizations using AWS Detective Controls generally reduce the time required to detect and respond to security incidents. The introduction of additional services allows for faster actions to mitigate threats compared to companies not using these tools. Moreover, AWS provides detailed analytics through tools such as AWS Security Hub, which offers convenient reports on security status and vulnerabilities, helping to promptly identify and resolve issues. GuardDuty, in turn, detects unusual activity in the account that could be a sign of possible breaches.

Using these tools can significantly enhance AWS account security, reduce the risk of data loss, and maintain business process stability, making them essential components of a modern cybersecurity strategy.

According to a study by IBM, the cost of detecting and eliminating cyberattacks increases by 13.7% annually on average [27]. However, according to the same study, the use of certain technologies and tools can reduce the cost of eliminating the consequences of cyberattacks from 2.7% to 9.7%.

One of the best practices for using these tools is to regularly check CloudTrail and VPC Flow Logs to identify potential security incidents and monitor network activity. In addition, it is recommended that you set up automatic subscriptions to Guard Duty notifications and regularly check the recommendations from Trusted Advisor to ensure optimal security for your account.

In summary, AWS users should pay close attention to their account security and utilize various tools and best practices to protect their accounts. Among other things, detective controls such as CloudTrail, VPC Flow Logs, Guard Duty, and Trusted Advisor can help identify potential security incidents and prevent their consequences.

In addition, it is recommended that you pay attention to setting up access to AWS resources, using multi-factor authentication, regularly updating passwords, and using data encryption.

2.1. Incident response to account loss and access compromise

Although the use of detective controls significantly reduces the risk of losing access to an AWS account or being hacked, such events can still happen. Therefore, it's essential to have a well-prepared incident response plan in place to minimize the consequences and quickly regain control over the account while protecting critical data and ensuring business continuity.

Accounts may be compromised for various reasons, including data theft involving login credentials such as usernames and passwords, often resulting from phishing attacks, social engineering, or targeted password attacks. Attackers may also attempt to access accounts by compromising personal keys used for authentication. Stolen or hacked keys can allow attackers to unauthorizedly log in to systems, retrieve confidential information, or even make dangerous changes to infrastructure or software.

Such incidents can lead to significant security breaches, risking confidential data leakage, disruption of critical processes, and damage to reputations. Attackers can exploit these vulnerabilities to alter system settings, modify software, or even block access to essential resources. These actions can result in major financial losses, penalties, a loss of customer trust, and operational delays.

To prevent such scenarios, it's essential not only to apply control mechanisms but also to have tools ready for a swift incident response. This includes clearly defined steps for identifying and isolating compromised accounts, resetting passwords and keys, and restoring the system to a stable state. Continuous employee training and the implementation of multi-factor authentication also play a key role in reducing risks.

Thus, even with strong security measures like AWS Detective Controls, having a comprehensive incident response plan is critical for companies to react quickly to threats, minimizing the negative impact on their operations.

The first step [28] in responding to account loss or access compromise is stopping the malicious activity. The following actions should be taken to ensure security:

- Change the password or access key: In case of suspicious activity on the account, it's necessary to immediately change the user's password or access key. After changing the password or key, all configuration files that use the old password or key should be updated.
- Disable the account: If there is suspicion that attackers have access to the account, it's critical to disable the account immediately to prevent further damage.
- Contact AWS Support: Upon detecting suspicious activities, users should contact AWS Support and

report possible account loss or compromised security [29].

In response to account loss or compromised access to AWS resources, Amazon Web Services provides an extensive set of tools to ensure system security and protection. After such an incident, it's important to gather detailed information about what occurred during a specific period across the entire system. This information will help administrators understand when and where the incident happened and assess its impact on systems and security. Elements of detective control can be used for a complete understanding of the events.

One of the most important tools is AWS CloudTrail, which provides detailed logs of user and resource activity within the AWS environment. The service stores activity logs, such as logins, logouts, resource creation and deletion, and data access. CloudTrail logs allow administrators to detect unusual activity, as well as verify and analyze potential security threats. With CloudTrail, you can track who performed what actions in your systems, which helps detect and investigate potential security incidents.

The use of AWS Detective Controls tools can significantly reduce the risk of account loss and unauthorized access to AWS resources. They provide automatic methods for configuring monitoring, auditing, and responding to security events. This allows companies and individual users to ensure maximum security for their accounts and resources within the AWS environment.

2.2. Detective controls for incident analysis

To prevent account loss in the AWS cloud platform and for incident analysis, AWS offers various detective controls. The detection tools provided by the AWS Cloud Platform play an important role in preventing account loss and analyzing security incidents. They help identify unusual or suspicious account activity that may indicate a potential security incident.

The key detective controls offered by AWS include:

- CloudTrail
- VPC Flow Logs
- Amazon GuardDuty
- AWS Trusted Advisor.

CloudTrail is an AWS-managed recording system that provides a detailed audit of actions performed in an account. The service logs all API calls, configuration changes, and user interactions, thereby creating a detailed trace for further analysis and reconstruction of the sequence of events. CloudTrail logs are stored in highly available Amazon S3 storage and can be analyzed using the AWS Management Console or third-party data analysis tools.

VPC Flow Logs is a service that records all network activity that occurs in an Amazon Virtual Private Cloud (VPC), such as incoming and outgoing packets over the network. AWS provides various ways to view VPC Flow Logs. Users can use the AWS VPC Flow Log Console or use third-party tools to analyze and visualize the data.

Amazon GuardDuty is a service that analyzes data from various sources, including CloudTrail activity logs and VPC Flow Logs, to identify potential security incidents. The service uses machine learning algorithms to detect unusual account activity, such as unauthorized access to account resources or configuration changes.

AWS Trusted Advisor is a service that provides recommendations to users to optimize costs, ensure high availability, and improve the security of their AWS infrastructure environment. The service uses data from users' accounts and analyzes it to make recommendations to optimize costs, ensure high availability, and improve the security of its infrastructure environment. Trusted Advisor offers several categories of recommendations, including security, performance, cost, and availability.

These controls allow administrators to detect potential threats and incidents at an early stage and take the necessary steps to prevent account loss and ensure the security of the AWS environment.

In addition to detecting threats and incidents, detective controls also help users gain more detailed information about these events. This can involve changing access settings, performing additional security checks, and collaborating with AWS Support for further investigation and incident response. It's important to note that detective controls are part of a broader range of security tools provided by AWS. By combining various tools and following best security practices, organizations and companies can enhance the security of their AWS environments and protect their accounts and data.

2.3. Using infrastructure as code to simplify the implementation of detective controls

With the introduction of AWS cloud services, the use of Infrastructure as Code (IaC), or automation tools such as Terraform, is becoming increasingly popular [30]. These tools allow users to manage and scale AWS infrastructure using code [31].

According to a 2021 analysis by Emergen Research, the market size for infrastructure programming tools reached $0.64 billion, and it is expected to grow at a compound annual growth rate (CAGR) of 240% over the next ten years [32]. Revenue is projected to grow from $0.64 billion in 2021 to $4.45 billion by 2030. The main driver of this growth is the demand for better optimization of business operations, necessitating new technological approaches as software systems become increasingly complex and advanced.

Terraform is a tool for automating the deployment of computational infrastructure, which can be used to automate the configuration and management of AWS infrastructure. Terraform supports a wide range of platforms, from major cloud providers like AWS, Azure, and GCP, to smaller platforms such as Hetzner or 1&1. Additionally, it works with software such as Docker, Kubernetes, and Chef, extending its functionality and allowing users to work with these in tandem.

Using Terraform, administrators can create and manage various AWS services, establish security policies, and respond to incidents quickly. One of the key benefits of using Terraform for deploying and managing AWS services is the ability to store the entire infrastructure in code. This simplifies the replication and scaling of infrastructure while ensuring security and preventing human errors. Additionally, Terraform makes it easy to manage more complex infrastructures, including networks, databases, and other services. Tasks addressed with Terraform and other infrastructure automation tools include reducing the risk of human error, efficiently managing infrastructure configuration, and minimizing the time required to provision and scale infrastructure while ensuring compliance with security and regulatory standards.

This and much more make Terraform a powerful tool for managing infrastructure, both within a single hosting platform and across multiple platforms simultaneously.

In summary, the use of Terraform or other infrastructure automation tools can help administrators more efficiently and securely manage their company's AWS infrastructure, reduce deployment time, minimize errors, and ensure compliance with security and regulatory standards.

3. Security as code—overview

Security as Code is a partially new approach that integrates security practices into the software development lifecycle by treating security configurations and policies as code. This methodology enables teams to automate and version control security measures alongside application code, ensuring that security is built-in from the start rather than tacked on later. By embedding security checks into Continuous Integration/Continuous Deployment (CI/CD) pipelines, organizations can proactively identify and remediate vulnerabilities before they reach production. This shift not only fosters a culture of security awareness among developers but also aligns security objectives with business goals, resulting in a more resilient and secure software ecosystem.

3.1. Advantages of security as code

One of the primary advantages of Security as Code is the enhancement of overall security posture through automation. By automating security checks, organizations can reduce the risk of human error and ensure consistent enforcement of security policies across all environments. Additionally, this approach enables faster feedback loops, allowing developers to detect and address vulnerabilities early in the development process, thereby minimizing costly post-release fixes. Furthermore, version control for security configurations facilitates easier auditing and compliance, as teams can track changes and maintain a clear history of security measures. Ultimately, Security as Code promotes collaboration between development and security teams, fostering a more agile and efficient approach to software development while significantly improving security outcomes.

3.2. Disadvantages of security as code

Despite its benefits, Security as Code also presents challenges. One significant disadvantage is the potential for over-reliance on automation, which can lead to complacency among teams. If security checks are automated without proper oversight, critical vulnerabilities may go unnoticed, resulting in security gaps. Additionally, implementing Security as Code requires a cultural shift and the acquisition of new skills, which can be daunting for some organizations. The initial setup of automated security tools can also be resource-intensive, requiring time and investment that may not yield immediate returns. Moreover, integrating these practices into existing workflows may introduce complexity, making it essential to carefully manage the transition to ensure it doesn't disrupt development processes.

3.3. Difference between manual and automated configuration

Manual configuration involves human intervention to set security policies and controls, which can be time-consuming and prone to human error. Each configuration change requires individual attention, leading to inconsistencies and potential security risks if best practices are not consistently applied. On the other hand, automated configuration employs tools and scripts to enforce security policies systematically and consistently. This approach reduces the likelihood of human error, enhances speed and efficiency, and allows for scalability in managing security across large environments. While manual configuration can offer flexibility in unique scenarios, automated configuration ensures a more uniform and reliable implementation of security measures, critical for maintaining a strong security posture in rapidly evolving software development landscapes.

4. Automating the configuration of some detective controls with Terraform

As we mentioned earlier, manually configuring these services is not a complicated process, the use of software-based configuration using Terraform is becoming more common due to the need for flexibility and re-deployment capabilities. The code samples provided below provide the basic structure for the setup of each detective control mentioned in this research with a brief explanation of each step and configuration.

4.1. Using infrastructure as code to set up AWS GuardDuty

```
resource "aws_guardduty_detector" "_" {
  enable = true
}
resource     "aws_guardduty_organization_admin_account"
"account_id" {
  admin_account_id = "123456789012"
}
resource "aws_guardduty_invite_accepter" "detector" {
  detector_id = aws_guardduty_detector._.id
  master_account_id                                    =
aws_guardduty_detector.primary.account_id
}
resource        "aws_guardduty_organization_configuration"
"example" {
  auto_enable_organization_members = "ALL"
  detector_id = aws_guardduty_detector._.id
  datasources {
    s3_logs {
      auto_enable = true
    }
    kubernetes {
      audit_logs {
        enable = true
      }
    }
    malware_protection {
      scan_ec2_instance_with_findings {
        ebs_volumes {
          auto_enable = true
        }
      }
    }
  }
}
resource "aws_guardduty_detector" "_" {
  provider = aws._
}
```

This GuardDuty configuration enables comprehensive threat detection across an AWS organization.

The *aws_guardduty_detector* resource activates GuardDuty for the entire organization, allowing it to monitor and detect potential security threats.

To streamline the process, the *aws_guardduty_invite_accepter* automatically accepts GuardDuty invitations for member accounts, ensuring centralized control over the organization's security.

The configuration part ensures that GuardDuty is automatically enabled for all current and future organization members, while also enabling advanced data source monitoring. This includes automatic logging for S3 access, Kubernetes audit logs for tracking resource access and changes, and malware protection scans on EC2 instances, particularly focusing on EBS volumes. Finally, a secondary *aws_guardduty_detector* resource configures another detector for use by the primary account for full security coverage.

4.2. Using infrastructure as code to set up AWS cloud trail

```
resource "aws_s3_bucket" "_" {
  bucket = "cloudtrail-bucket"
  versioning {
    enabled = true
  }
}

resource "aws_cloudtrail" "CT" {
  name                          = "test-cloudtrail"
  s3_bucket_name                = aws_s3_bucket._.bucket
  include_global_service_events = true
  is_multi_region_trail         = true

  event_selector {
    read_write_type = "All"
    include_management_events = true
    data_resource {
      type   = "AWS::S3::Object"
      values = ["arn:aws:s3:::cloudtrail-bucket/*"]
    }
  }
}
```

This CloudTrail configuration sets up comprehensive logging for AWS account activities. The *aws_s3_bucket* resource creates an S3 bucket named "cloudtrail-bucket", which is used to store CloudTrail logs, with versioning enabled to protect log integrity.

The *aws_cloudtrail* resource establishes a CloudTrail named "test-cloudtrail" that logs all API activity and events across AWS services. It includes global service events, making it a multi-region trail that captures actions in all AWS regions.

An *event_selector* is configured to log all read and write operations, and it ensures that management events, such as account-level changes, are captured.

The configuration also specifies that all S3 object-level activity within the "cloudtrail-bucket" is logged, ensuring detailed audit trails for S3 operations.

4.3. Using infrastructure as code to set up VPC flow logs

```
# Create VPC
resource "aws_vpc" "test_vpc" {
  cidr_block = "10.0.0.0/16"
}

# Create subnet
resource "aws_subnet" "test_subnet" {
  vpc_id                  = aws_vpc.test_vpc.id
  cidr_block              = "10.0.1.0/24"
  availability_zone       = "us-west-1a"
}

# Create flow log
resource "aws_flow_log" "test_flow_log" {
  iam_role_arn    = aws_iam_role.test_flow_log_role.arn
  traffic_type    = "ALL"
  log_destination             =             "arn:aws:logs:us-west-
1:123456789012:log-group:/aws/vpc/flow-logs"

  vpc_id          = aws_vpc.test_vpc.id
  subnet_id       = aws_subnet.test_subnet.id
}

# Create IAM role for flow logs
resource "aws_iam_role" "test_flow_log_role" {
  name = "test-flow-log-role"
  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "vpc-flow-logs.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
}

# Attach necessary policies to the IAM role
resource "aws_iam_role_policy_attachment" "_" {
  role       = aws_iam_role.test_flow_log_role.name
  policy_arn          =          "arn:aws:iam::aws:policy/service-
role/AmazonVPCFullAccess"
}
```

This VPC Flow Logs configuration is designed to capture and monitor network traffic within a defined Virtual Private Cloud (VPC) environment. The *aws_vpc* resource creates a VPC with a CIDR block of 10.0.0.0/16, which sets up the network's IP address range and provides an isolated section within the AWS infrastructure for deploying resources. Within this VPC, the *aws_subnet* resource provisions a subnet, defining a smaller CIDR block of 10.0.1.0/24 within the VPC's range, and placing it in the availability zone us-west-1a for high availability and redundancy.

The *aws_flow_log* resource is used to enable VPC Flow Logs, which captures all network traffic (traffic_type = "ALL") both to and from resources within the VPC and subnet. The logs are sent to a specified CloudWatch log group (arn:aws:logs:us-west-1:123456789012:log-group:/aws/vpc/flow-logs), allowing for real-time monitoring and analysis of network traffic. This is crucial for security, troubleshooting, and compliance purposes, as it allows the tracking of all VPC activity, including incoming and outgoing data.

To facilitate the logging, an IAM role is created with *aws_iam_role*, named "test-flow-log-role", which includes an assume-role policy allowing the VPC Flow Logs service to take on this role. The policy grants the necessary permissions for the VPC Flow Logs service to write log data into CloudWatch. Finally, the *aws_iam_role_policy_*

attachment attaches the AmazonVPCFullAccess policy to the IAM role, providing full permissions over VPC resources and flow log management. This ensures that the role can manage and interact with the flow of log resources effectively while allowing logs to be securely stored and accessed for further analysis.

4.4. Using infrastructure as code to set up AWS trusted advisor

```
#Trusted Advisor

# Enable Trusted Advisor checks
resource "aws_trusted_advisor_check_refresh_status" "example" {
  check_ids = [
    "eW7HH0l5z0"  # Replace with the check IDs you want to enable
  ]
}

# Enable SNS notifications for Trusted Advisor
resource "aws_sns_topic" "example" {
  name = "trusted-advisor-notifications"
}

resource "aws_trusted_advisor_check" "example" {
  check_id  = "eW7HH0l5z0"  # Replace with the check ID you want to enable
  sns_topic = aws_sns_topic.example.arn
}
```

This Trusted Advisor configuration automates the process of enabling, refreshing, and monitoring AWS Trusted Advisor checks. The *aws_trusted_advisor_check_refresh_status* resource ensures that specific Trusted Advisor checks, identified by their check_ids (in this case, "eW7HH0l5z0"), are refreshed regularly to provide up-to-date insights on the health and best practices of the AWS environment. Trusted Advisor checks cover key areas like cost optimization, performance, security, and fault tolerance, making it essential to have the latest data available.

To facilitate real-time notifications, the *aws_sns_topic* resource creates an SNS (Simple Notification Service) topic, named "trusted-advisor-notifications", which is used to broadcast alerts or updates related to Trusted Advisor checks. This ensures that any critical issues or improvements recommended by the Trusted Advisor can trigger immediate notifications to the relevant stakeholders.

The *aws_trusted_advisor_check* resource configures a specific check (check_id = "eW7HH0l5z0") and ties it to the previously created SNS topic. By linking the check with SNS, any changes in the status of this Trusted Advisor check will automatically trigger notifications, keeping the team informed of critical findings in real-time.

5. Conclusions

This study focused on the use of key AWS security tools: CloudTrail, VPC Flow Logs, GuardDuty, and Trusted Advisor. Each of these tools provides specific security controls to monitor and prevent unauthorized access to AWS infrastructure.

While manual configuration allows us to quickly access AWS Detective Controls services, it requires significant manual effort and maintaining consistency across environments. Instead, infrastructure as code using Terraform provides a repeatable and scalable configuration of AWS Detective Controls. This approach simplifies the configuration and management of AWS services with a convenient, easy-to-understand configuration file format that is stored in a repository.

The main objective of the study was to examine in detail the capabilities of AWS Detective Controls to detect and prevent security threats in cloud environments. To effectively deploy and manage these mechanisms, we used Infrastructure as a Code (IaC) with Terraform. Thanks to Terraform modules, we achieved a stable and auditable AWS Detective Controls configuration, which allowed us to focus on analyzing and optimizing threat detection rules.

References

[1] D. Tykholaz, R. Banakh, Account Protection in AWS with Detective Controls, in: International Scientific and Technical Conference (2023) 95–96.

[2] V. Khoma, et al., Comprehensive Approach for Developing an Enterprise Cloud Infrastructure, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3654 (2024) 201–215.

[3] O. Vakhula, I. Opirskyy, O. Mykhaylova, Research on Security Challenges in Cloud Environments and Solutions based on the "Security-as-Code" Approach, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3550 (2023) 55–69.

[4] S. Vasylyshyn, et al., A model of decoy system based on dynamic attributes for cybercrime investigation, Eastern-European J. Enterprise Technol. 1(9(121) (2023) 6–20. doi: 10.15587/1729-4061.2023.273363.

[5] S. Yevseiev, et al., Models of socio-cyber-physical systems security, Kharkiv: PC TECHNOLOGY CENTER, 184 (2023). doi: 10.15587/978-617-7319-72-5.

[6] Most enterprises highly vulnerable to security events caused by cloud misconfiguration. URL: https://www.helpnetsecurity.com/2018/10/05/cloud-misconfiguration/

[7] IBM Report: Escalating Data Breach Disruption Pushes Costs to New Highs. URL: https://newsroom.ibm.com/2024-07-30-ibm-report-escalating-data-breach-disruption-pushes-costs-to-new-highs

[8] Critical MOVEit vulnerability puts huge swaths of the Internet at severe risk. URL: https://arstechnica.com/security/2024/06/critical-moveit-vulnerability-puts-huge-swaths-of-the-internet-at-severe-risk/

[9] Boeing says "cyber incident" hit parts business after ransom threat URL: https://www.reuters.com/business/aerospace-defense/boeing-investigating-cyber-incident-affecting-parts-business-2023-11-01/

[10] AT&T Addresses Recent Data Set Released on the Dark Web. URL: https://about.att.com/story/2024/addressing-data-set-released-on-dark-web.html

[11] Bank of America 2024 Data Breach and Third Party Risk. URL: https://panorays.com/blog/boa-data-breach-2024/

[12] Verizon's 2024 Data Breach Investigations Report (DBIR). URL: https://www.verizon.com/business/resources/reports/dbir/2024/summary-of-findings/

[13] Y. Dreis, et al., Model to Formation Data Base of Internal Parameters for Assessing the Status of the

State Secret Protection, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, CPITS, vol. 3654 (2024) 277–289.
[14] P. Anakhov, et al., Protecting Objects of Critical Information Infrastructure from Wartime Cyber Attacks by Decentralizing the Telecommunications Network, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 3050 (2023) 240-245.
[15] A. Zahynei, et al., Method for Calculating the Residual Resource of Fog Node Elements of Distributed Information Systems of Critical Infrastructure Facilities, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, CPITS, vol. 3654 (2024) 432–439.
[16] Multi-Factor Authentication (MFA) for IAM. URL: https://aws.amazon.com/iam/features/mfa/
[17] IAM. URL: https://aws.amazon.com/iam/?nc=bc&pg=f-mfa
[18] P. Anakhov, et al., Evaluation Method of the Physical Compatibility of Equipment in a Hybrid Information Transmission Network, Journal of Theoretical and Applied Information Technology 100(22) (2022) 6635–6644.
[19] V. Zhebka, et al., Optimization of Machine Learning Method to Improve the Management Efficiency of Heterogeneous Telecommunication Network, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 3288 (2022) 149–155.
[20] V. Zhebka, et al., Stability Method of Connectivity Automated Calculation for Heterogeneous Telecommunication Network, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 3188 (2021) 282–287.
[21] Routavaara, Security monitoring in AWS public cloud, Bachelor's Thesis, Technology Information and Communication Technology (2020).
[22] Detective controls. URL: https://docs.aws.amazon.com/prescriptive-guidance/latest/aws-security-controls/detective-controls.html
[23] CloudTrail. URL: https://aws.amazon.com/cloudtrail/?nc1=h_ls
[24] VPC Flow Logs – Log and View Network Traffic Flows. URL: https://aws.amazon.com/blogs/aws/vpc-flow-logs-log-and-view-network-traffic-flows/
[25] Guard Duty. URL: https://www.amazonaws.cn/en/guardduty/
[26] Trusted Advisor. URL: https://docs.aws.amazon.com/awssupport/latest/user/trusted-advisor.html
[27] IBM Cost of data breaches 2022. URL: https://www.ibm.com/reports/data-breach
[28] AWS Security Incident Response. URL: https://docs.aws.amazon.com/whitepapers/latest/aws-security-incident-response-guide/aws-security-incident-response-guide.html
[29] AWS Support. URL: https://console.aws.amazon.com/support/home/?nc2=h_ql_cu
[30] Terraform. URL: https://www.terraform.io/
[31] Terraform usage. URL: https://enlyft.com/tech/products/hashicorp-terraform
[32] Infrastructure as Code Market Size to Reach USD 4.45 Billion in 2030 | Emergen Research. URL: https://www.prnewswire.com/news-releases/infrastructure-as-code-market-size-to-reach-usd-4-45-billion-in-2030--emergen-research-301737553.html