November 10, 2023

MEDCIRCLE VULNERABILITY ASSESSMENT REPORT

Pranay Venkata Bhamidipati(pbhamid1); Kushangi Nilpeshbhai Patel (kpatel64); James Graves (jgrav008)

Table of Contents

I. Vulnerability assessment report

A. Identified weaknesses, impact, and recommended mitigations

1. High severity vulnerabilities

a. Excessive Permissions
   i. Vulnerability: Complete access to the S3 bucket with "Action": "s3:*" is permitted by the IAM user's policy. This is rather liberal since it gives the user complete control over the S3 bucket.
   ii. Recommendations:
      1. Restricting rights to the absolute minimum required for the user's particular tasks will help to mitigate this.
      2. Just use resources and actions that are truly necessary for the user.
      3. When (Amazon Web Services, 2023) (Microsoft Inc., 2023) (Brush, 2022)describing the necessary permissions, be precise and refrain from using "Action": "s3:*".
      4. Put in place robust access controls to safeguard the S3 bucket. (Amazon Web Services, 2023)

b. Misconfiguration of the DBPassword Security Group
   i. Vulnerability: Improper DBPassword parameter setup involving an unnecessary EC2 instance type constraint.
   ii. Recommendation: The DBPassword parameter's needless constraint be removed.

c. SSHLocation Criteria
   i. Vulnerability: The default SSH IP address, 0.0.0.0/0, is wide-ranging and vulnerable.
   ii. Recommendation: Limit the use of the SSHLocation argument to CIDR blocks or trustworthy IP addresses only. Save the use of 0.0.0.0/0 for emergencies only.

d. Groups for Public and Private Security
   i. Vulnerability: Permitting unlimited access to specific ports.
   ii. Recommendation: restrict IP ranges in security group rules so that they can only be accessed by reliable sources. Observe the least privileged concept.

e. Extremely Weak Admin and Developer Guidelines
   i. Vulnerability: The excessively lax policies of the Admin and Developer roles permit any action on any resource, posing a serious security risk.
   ii. Recommendation:
      1. Limit the resources and actions that can be performed by Admin and Developer roles.
      2. To reduce the attack surface, adhere to the least privilege concept.
      3. Review and improve these policies frequently to get rid of permits that aren't needed.

2. Medium severity vulnerabilities

a. Access Key Creation
   i. Vulnerability: The user's IAM access key is created by the template. Access keys need to be stored with caution. It presents a potential security risk even if it isn't a direct vulnerability.

ii. Recommendations:
    1. Analyze if the use case calls for the creation of an IAM access key. Do not make more access keys if not needed.
    2. For improved access key security, rotate the access keys frequently and make use of AWS Secrets Manager or other services. (Microsoft Inc., 2023)

b. Bucket Name Generation
    i. Vulnerability: The S3 bucket name in the template contains the AWS account ID. This might make the AWS account ID visible, which raises privacy issues. Although it's not a security flaw, privacy is something to think about.
    ii. Recommendation: If revealing the AWS account ID in the S3 bucket name raises privacy concerns then use different naming patterns that omit sensitive data.
c. Password for DBInstance
    i. Vulnerability: The CloudFormation template stores database passwords in cleartext.
    ii. Recommendation: Use AWS Secrets Manager or another more secure method to handle database credentials.
d. SecurityGroup Instance and SecurityGroup RDSSecurity
    i. Vulnerability: Opening up particular ports for access without specifically naming reliable sources.
    ii. Recommendation: Strictly limit access to CIDR blocks or trusted IP ranges, and put rigorous security group restrictions in place.
e. Unrestricted Access to S3 Bucket for Guests
    i. vulnerability: A policy on the Guest role permits s3: GetObject on all objects in the "public bucket." Sensitive information could be accessed by abusing this privilege.
    ii. Recommendation: It is suggested that the Guest's role's rights be restricted to the particular items or activities that it requires. Limit access to the necessary resources in order to reduce the possibility of misuse.
f. Using generic Usernames
    i. Vulnerability: "Patient1," "Patient2," and similar generic usernames are used by the IAM users. It might be simpler for attackers to identify or target certain users if usernames are predictable or generic.
    ii. Recommendation: To improve security, use usernames that are more complex and less predictable.

3. Low severity vulnerabilities

a. Bucket encryption
    i. Vulnerability: It's a good practice to configure server-side encryption using AES256 for the S3 bucket. But encrypted data should also be transferred. This is not so much a vulnerability as it is an excellent practice.
    ii. Recommendation: Encrypt data while it's in transit to and from the S3 bucket in addition to using server-side encryption with AES256. When connecting to the S3 bucket via HTTPS, think about turning on encryption while in transit.
b. Lack of logging and monitoring

i. Vulnerability: S3 bucket logging and CloudWatch logging are not configured in the template to track S3 activity. The lack of these functionalities is less of a direct risk and more of a configuration gap, even though security is still crucial.
ii. Recommendations:
    1. In order to identify and address possible security incidents, log and monitor S3 operations.
    2. Set up S3 bucket logging to record events, then connect it to AWS CloudWatch for oversight.
    3. In the event of unwanted or questionable access, set up notifications and alarms.

c. Subnet Public
    i. Vulnerability: Creating a public subnet and possibly exposing resources to the public internet by setting MapPublicIpOnLaunch to True.
    ii. Recommendation: Evaluate if public IP addresses are necessary for the resources in the public subnet. If not, to improve security, set MapPublicIpOnLaunch to False.
d. IAM Policy Is Weak
    i. Vulnerability: EC2 instances and resources lack clearly specified IAM roles or policies.
    ii. Recommendation: It is advised that IAM roles and policies be defined and attached to resources while adhering to the least privileged concept.
e. Inadequate Version Control
    i. Vulnerability: There is no version control in the IAM policies, which makes it difficult to monitor upgrades and changes over time.
    ii. Recommendation: It is suggested that version control be added to IAM policies by indicating the version of the policy and documenting any modifications made to it. Auditing and policy management will benefit from this.
f. Lack of Group and Policy attachments to Users
    i. Vulnerability: No IAM groups or policies are assigned to the newly generated IAM users. This could make it difficult to scale and manage permissions in an orderly fashion.
    ii. Recommendations:
    1. Create IAM groups and policies, then attach users to these groups.
    2. Make use of groups and policies to create centralized control and simplify policy updates.

II. Data security assessment report

A. Identified weaknesses, impact, and recommended mitigations

1. High severity vulnerabilities

There were no high severity items identified.

2. Medium severity vulnerabilities

a. InstanceSecurityGroup SecurityGroupIngress:

i. Vulnerability: From any IP address (CidrIp: 0.0.0.0/0), the InstanceSecurityGroup permits unfettered access to port 80 (FromPort: '80', ToPort: '80'). This might pose a security threat.
ii. Recommendation: Limit IP ranges that can access port 80 to only those that is known to be trustworthy, like known client IP ranges. Steer clear of giving vulnerable ports like port 80 unrestricted access.

b. SSH Access InstanceSecurityGroup SecurityGroupIngress:
i. Vulnerability: Unrestricted SSH access is permitted by the InstanceSecurityGroup (FromPort: '22', ToPort: '22', CidrIp:!Ref SSHLocation). This exposes SSH access to the public internet, which is a security issue.
ii. Recommendation: use an isolated host or limit SSH access to particular trusted IP ranges in order to ensure safe SSH access.

c. SecurityGroupIngress, RDSSecurityGroup:
i. Vulnerability: From the 10.1.10.0/24 CIDR range, port 3306 (FromPort: '3306', ToPort: '3306') is open to all users via the RDSSecurityGroup. This arrangement might not be the safest one.
ii. Recommendation: Restrict access to port 3306 to only trusted IP ranges that need access to the RDS database. Review and update the security group rules as necessary.

3. Low severity vulnerabilities

a. Description of DBPassword Constraint:
i. Vulnerability: Two ConstraintDescription properties on the DBPassword argument are not permitted in CloudFormation. This is unnecessary and may cause confusion.
ii. Recommendation: use a single, unambiguous description of the constraint and remove one of the ConstraintDescription properties in order to prevent confusion.

b. PublicSecurityGroup GroupDescription:
i. Vulnerability: Two GroupDescription attributes are present in the PublicSecurityGroup resource, which is not permitted in CloudFormation.
ii. Recommendation: use a single, unambiguous description for the security group and to remove one of the GroupDescription properties in order to prevent confusion.

B. CloudFormation Vulnerabilities

1. The CloudFormation template creates an S3 bucket and an IAM user with access to the bucket for storing data. The IAM user has broad permissions (`s3:*`) on the specified bucket.
i. Vulnerability: - Unencrypted Data. The template does not explicitly enable server-side encryption for the S3 bucket. This poses a significant security risk, as sensitive patient data may be stored without encryption. In the event of unauthorized access or breaches, this data would be exposed.
ii. Recommendations:
1. Enable Server-Side Encryption:

Modify the CloudFormation template to ensure that server-side encryption is enabled for the S3 bucket. You can specify the encryption method as AES256,

which is a robust encryption standard. This provides an additional layer of security for data at rest.

2. Implement Fine-Grained Access Controls:

   Review and refine the IAM policies and permissions. Assign permissions at the least privilege principle, so that the IAM user has only the necessary access required for specific operations. Limit permissions to read/write access to the designated bucket.

3. Regularly Monitor and Audit Access:

   Implement a robust monitoring and auditing mechanism to track user activities. AWS CloudTrail and S3 access logs should be enabled. Regularly review these logs to identify any suspicious activities or unauthorized access.

2. Potential vulnerabilities in the provided CloudFormation template from a data security perspective:
   i. Unencrypted Data in Transit:

      The template does not explicitly mention enabling SSL/TLS for MySQL connections (`DBInstance` resource). Without encryption, data transmitted between the application and the database could be intercepted.

   ii. Unencrypted Data at Rest:

      The template does not enable encryption at rest for the RDS instance (`DBInstance` resource). Storing sensitive patient data without encryption poses a risk in case of unauthorized access to the underlying storage.

   iii. SSH Access Security:

      The SSH security group (`PublicSecurityGroup`) allows unrestricted access (`0.0.0.0/0`). This could lead to unauthorized access if SSH keys are compromised or if the security group is not properly configured.

C. Unencrypted data vulnerabilities and corresponding recommendations

i. Unencrypted Data in Transit:
   a. Vulnerability: The template does not explicitly mention enabling SSL/TLS for MySQL connections (`DBInstance` resource). Without encryption, data transmitted between the application and the database could be intercepted.
   b. Recommendation: Enable SSL/TLS for MySQL connections. Update the `DBInstance` resource with the necessary properties for SSL encryption.

ii. Unencrypted Data at Rest:
   a. Vulnerability: The database does not enable encryption at rest for the RDS instance (`DBInstance` resource). Storing sensitive patient data without encryption poses a risk in case of unauthorized access to the underlying storage.

b. Recommendation: Enable encryption at rest for the RDS instance using AWS Key Management Service (KMS). Set the `StorageEncrypted` property to `true` and specify a `KmsKeyId`.

D. Additional Vulnerabilities and recommendations

1. SSH Access Security:
   a. Vulnerability:The SSH security group (`PublicSecurityGroup`) allows unrestricted access (`0.0.0.0/0`). This could lead to unauthorized access if SSH keys are compromised or if the security group is not properly configured.
   b. Recommendation: Restrict SSH access to specific IP ranges or only allow access through a bastion host. Avoid using `0.0.0.0/0` in the security group rules.
2. Database Credentials Management:
   a. Vulnerability: The database username and password are hardcoded in the User (`DBUsername` and `DBPassword`). Storing credentials in plaintext poses a risk if the template is exposed or shared improperly.
   b. Recommendation: Use AWS Secrets Manager or AWS Systems Manager Parameter Store to securely store and manage database credentials. Avoid hardcoding credentials in the template.
3. Security Group Misconfigurations:
   a. Vulnerability: The security group for the RDS instance (`RDSSecurityGroup`) allows database access from the entire `10.1.10.0/24` subnet. This might grant unnecessary access to other resources in that subnet.
   b. Recommendation: Update the `RDSSecurityGroup` to allow access only from specific trusted sources.
4. KeyPair Management:
   a. Vulnerability: The management of the EC2 KeyPair for SSH access (`KeyName`) is crucial. If the key is lost or compromised, unauthorized access to instances is possible.
   b. Recommendation: Regularly rotate SSH keys and update the `KeyName` in the template. Follow best practices for key management and avoid sharing private keys.

III. Virtual machine vulnerability assessment report

A. Vulnerabilities in Virtual machines

Virtual machines may be deployed using AWS EC2. It's a cloud-based utility from AWS that lets MedCircle increase or decrease computing power as needed. Health-related software and data are hosted by the firm on them. Some of the issues found are described below.

B. Identified weaknesses, impact, recommended mitigations, and recommendations for patching and vulnerability management

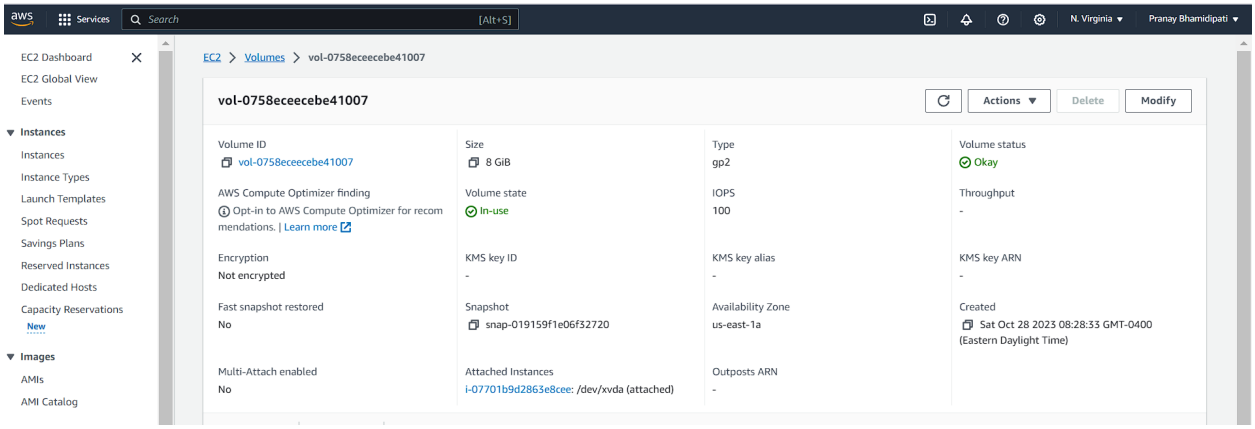
1. High severity vulnerabilities

a. Unencrypted EBS Volume

Description: unauthorized entities may get access to different data components. The absence of encryption poses a significant and immediate threat to the confidentiality, integrity, and availability of data. (White, 2020)

i. Vulnerabilities:

a. Unauthorized entities may gain access to different data components. The absence of encryption poses a significant and immediate threat to the confidentiality, integrity, and availability of data.



b. Data in idle state is at risk:

1. Potential access to confidential data.
2. All of the sensitive information is at risk of being compromised since it is not encrypted.

c. Data during transmission is at risk:

2. There is no encryption on the data that travels across EC2 instances and the associated EBS volumes.
3. When information is transferred across unsecured networks, this might pose a security risk since it increases the likelihood that it will be intercepted by malicious individuals.

d. The issue of non-compliance and its associated risks:

Assuming MedCircle company is a US based organization, not complying with Health Insurance Portability and Accountability Act (HIPPA)

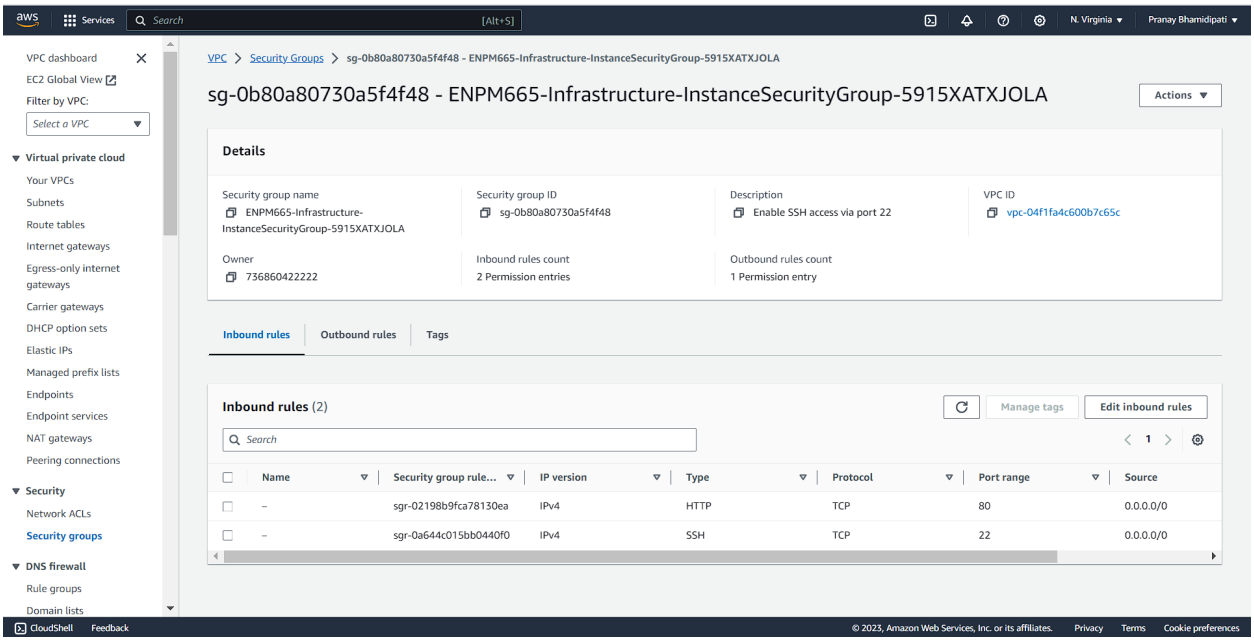
ii. Recommendations:

To mitigate these issues, (CloudYali Team, 2023) MedCircle should activate encryption for the Elastic Block Store (EBS) volumes, ensuring that encryption is implemented both during data storage and data transmission. Encrypting the EBS volumes serves to greatly bolster the security of the company's data, so safeguarding it against unwanted access and any breaches. Amazon Web Services (AWS) offers a range of features and tools that facilitate the implementation of encryption mechanisms. These include:

a. Switching on default encryption to Encrypt newly created EBS volumes should be performed automatically.

b. Encryption when data is idle:
   MedCircle has the option to use AWS Key Management Service for the purpose of encrypting EBS volumes while they are at rest. This guarantees that data is securely saved in an encrypted format.
c. Encryption during data transmission:
   During transmission of information, MedCircle may employ safe methods like SSH or HTTPS.

b. Security group SSH port is open
Amazon Web Services (AWS) resources, for example EC2 instances, (Jyl, 2023)may have their incoming and outgoing traffic governed by AWS Security Groups, which are virtual firewalls.



i. Vulnerabilities:
   a. The SSH port, (port 22 by default), is exposed to everyone on the internet.
   b. Any machine that is connected to the internet would be able to set up a secure shell (SSH) connections with the EC2 instance that is tied to it.
   c. It is possible this may lead to illegal access, incidents using brute force attacks, and additional breaches.
ii. Recommendations:
   a. Enable CloudWatch Logs and AWS CloudTrail facilitates the monitoring and logging of SSH access attempts, hence facilitating recognizing and handling of security events.
   b. The removal of inbound rules is a practice that upholds the concept of least privilege, guaranteeing that only essential ports and services remain available. This action effectively reduces the attack surface.

2. Medium severity vulnerabilities

a. Rulesets aren't empty for preset security groups

Description: The default security group permits unrestricted communication, hence potentially compromising security. The Amazon Elastic Computer Cloud (EC2) instances are accompanied by default security groups that include predefined policies.

i. Vulnerability: The presence of non-empty default security groups might result in the exposure of EC2 instances to undesired network traffic presumably originating from the public internet.

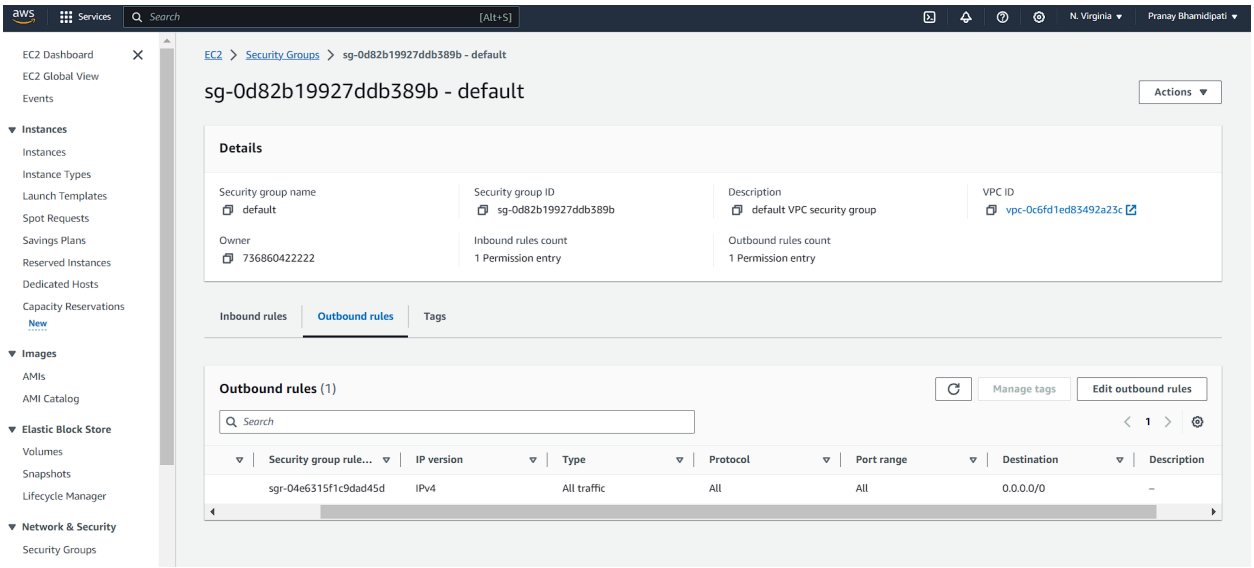
ii. Recommendation:

a. Conduct Periodic evaluations of the default security groups and verify that they do not possess any inbound or outbound rules.

b. Make the company's own security groups for EC2 instances instead of using predefined ones.

b. Security groups have unrestricted networks

Description: By design, all sources, ports, and protocols are unrestricted inside the security group, allowing for unrestricted communication across any instances belonging to the security group.



i. Vulnerability:

a. If instances inside the security group contain sensitive information, the absence of traffic restrictions might result in data disclosure or leakage.

b. It is possible for hostile or unauthorized users who have access to instances inside the security group to take advantage and elevate their privileges, do illegal acts, or commit other forms of harm.

ii. Recommendation:

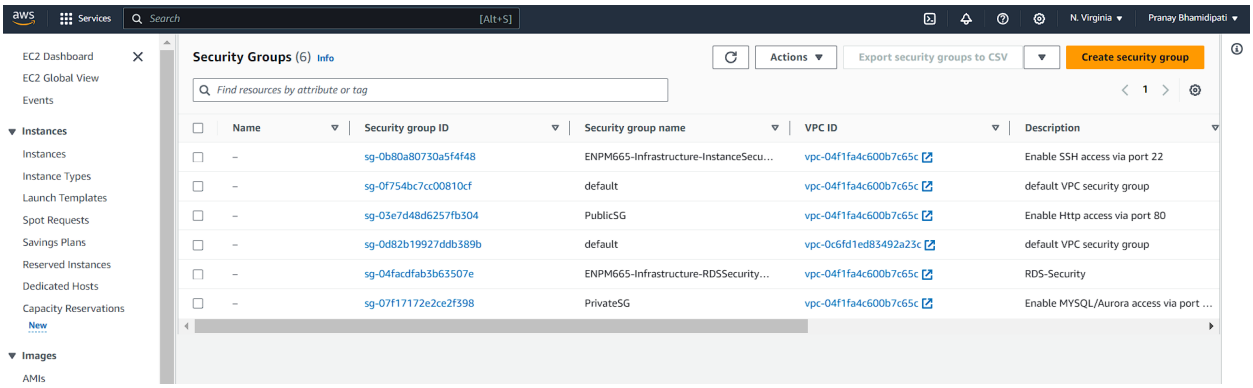
a. The rules for incoming and outgoing traffic should be restricted to enable it only from authorized sources and for specified reasons. Rules that utilize IP address ranges like 0.0.0.0/0 should be avoided.

b. By using AWS Virtual Private Cloud (VPC's) and subnets, MedCircle can incorporate segmentation to separate the company's network into several security groups to prevent data exchange among instances which don't require full access to each other.

3. Low severity vulnerabilities

a. Security group that is not used:

Description:
A security group that was formerly linked to AWS resources but is now disconnected. These inactive security groups may pose a threat and should be removed.



i. Vulnerability: Misconfigurations might occur if unused security groups were kept unmodified and had rules that applied to resources that are no longer in use.

ii. Recommendation:
   a. Conduct periodic inspections in order to locate security groups that have are no longer connected to any operational AWS services.
   b. It is recommended to check the inactive security groups and delete them if they are no longer necessary.

IV. Network security assessment report

AWS employs a multi-layered approach to network security which includes Network Access Control Lists (NACL's), Virtual Private Cloud (VPC's), security groups, and other components. (Amazon Web Services, 2023) AWS VPCs provide MedCircle the ability to construct isolated and logically distinct portions of the AWS cloud inside which they may

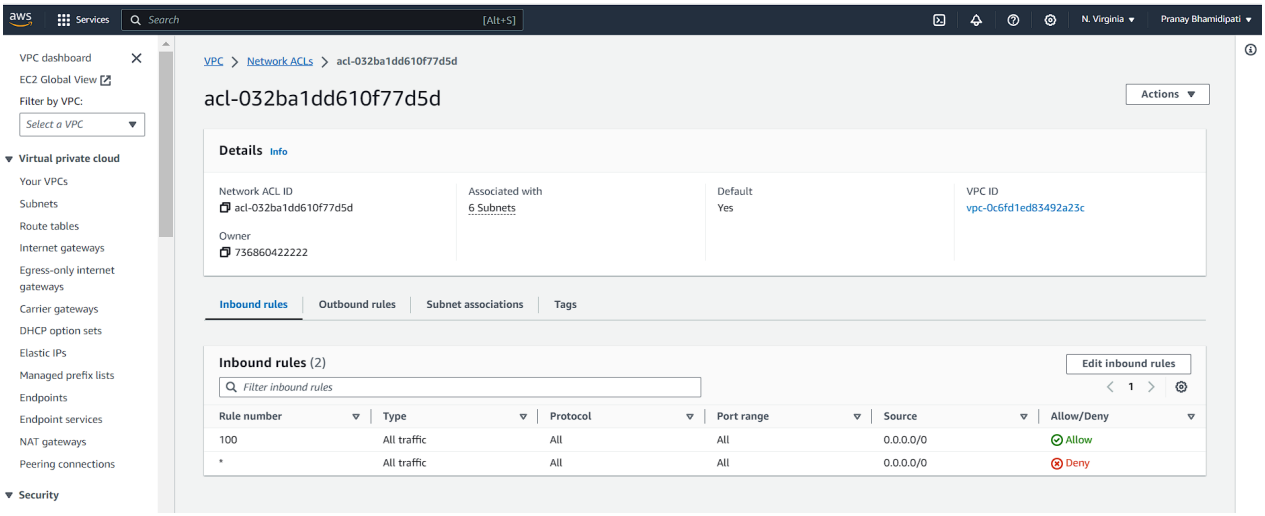
deploy services in a number of different subnets. The AWS NACL functions essentially as stateless network-level firewalls. With the help of AWS NACl, MedCircle could generate NACL rules that can regulate incoming and outgoing traffic at the subnet level.

1. High severity vulnerabilities

a. Default NACL setting to Allow All ingress traffic.

Description:

The Network Access Control List (NACL) in Amazon Web Services (AWS) is set up with an “Allow All" setting for the ingress traffic. (Adkoli, 2018) This configuration implies that there are no limitations imposed on the nature of the traffic, its source, or its destination. This setup enables unrestricted network connectivity between and within those linked subnets.



i. Vulnerability:

Enabling unrestricted ingress of all traffic might potentially expose valuable resources to substantial security risks, since it fails to impose any limitations on the types of traffic that are permitted. This has the potential to result in illegal access and possible security breaches within MedCircle. The most common attack can be Denial of Service (DoS) Attack.

ii. Recommendation:

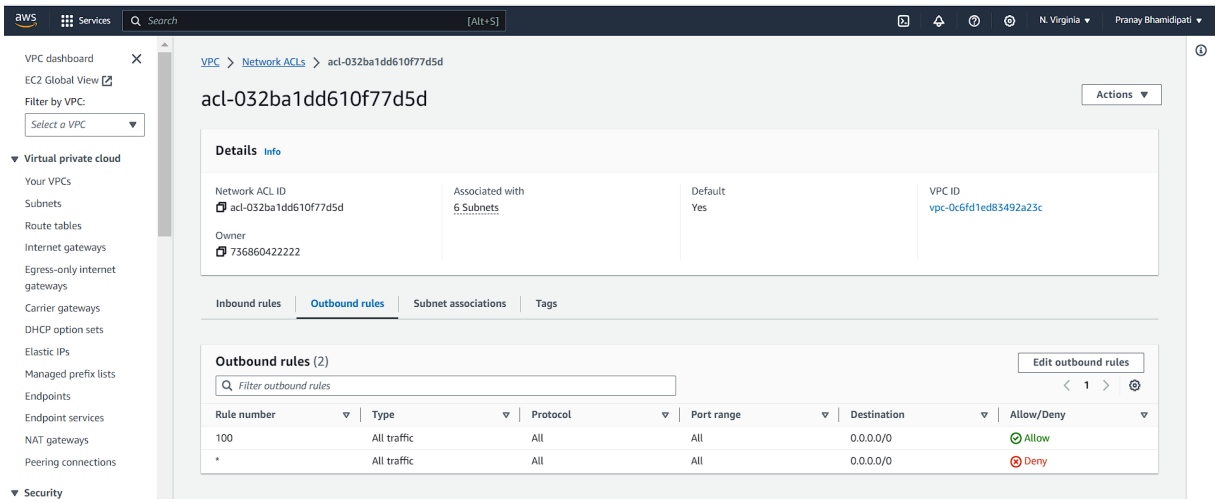
a. MedCircle should figure out what kinds of incoming traffic the company's services and applications need and add rules to the NACL that only let the required traffic through and stop all other traffic. Example, only let certain ports to run for certain services.

b. Conduct routine audits and reviews of the NACL rules to ensure that they continue to meet the company's security needs. Modify components as needed in order to adapt to altering demands of the services and applications.

b. Default NACL setting to allow all egress traffic:

Description:

The “Allow All" setting for the egress traffic implies that there are no limitations imposed on the nature of the traffic, its source, or its destination. This setup enables unrestricted network connectivity between and within those linked subnets.



i. Vulnerabilities:

a. Enabling unrestricted egress traffic might potentially expose valuable resources to substantial security risks, since it fails to impose any limitations on the types of traffic that are going out.

b. Enabling unregulated egress traffic has the potential to facilitate the unauthorized transfer of data, leading to the transfer of critical information from the company's network without sufficient security measures in place.

ii. Recommendations:

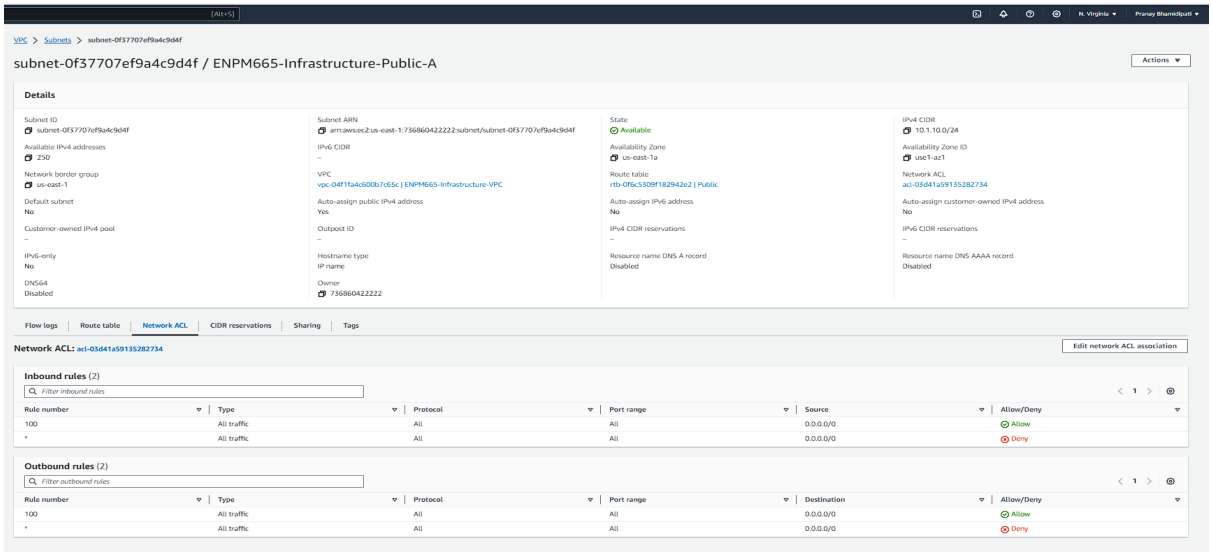
a. Determine the types of outgoing traffic the company's services and applications need. (Hacker News, 20233) Add rules to the NACL that only let the required traffic go out and stop all other traffic.

b. Conduct routine audits and reviews of the NACL rules to ensure that they continue to meet the company's security needs. Modify components as needed in order to adapt to altering demands of the services and applications.

c. NACL subnet setting to Allow All egress and ingress traffic:

Description:

The subnet allows all outgoing and incoming communication. (Lee, 2019) This comprises traffic of any kind of protocol and port number. This is a kind of insecure setup where all network traffic is permitted to come and go out of the network.



i. Vulnerabilities:

Improperly configured NACL subnets may put all resources on a subnet at risk. Hence, due to these exposed settings, hackers might exploit the company’s network to propagate malware or carry out harmful operations, putting your data at risk.

ii. Recommendations:

a. Determine the types of outgoing and incoming traffic the company's services and applications need for the specific subnets. Add rules to the NACL subnets that only let the required traffic go out and come in.

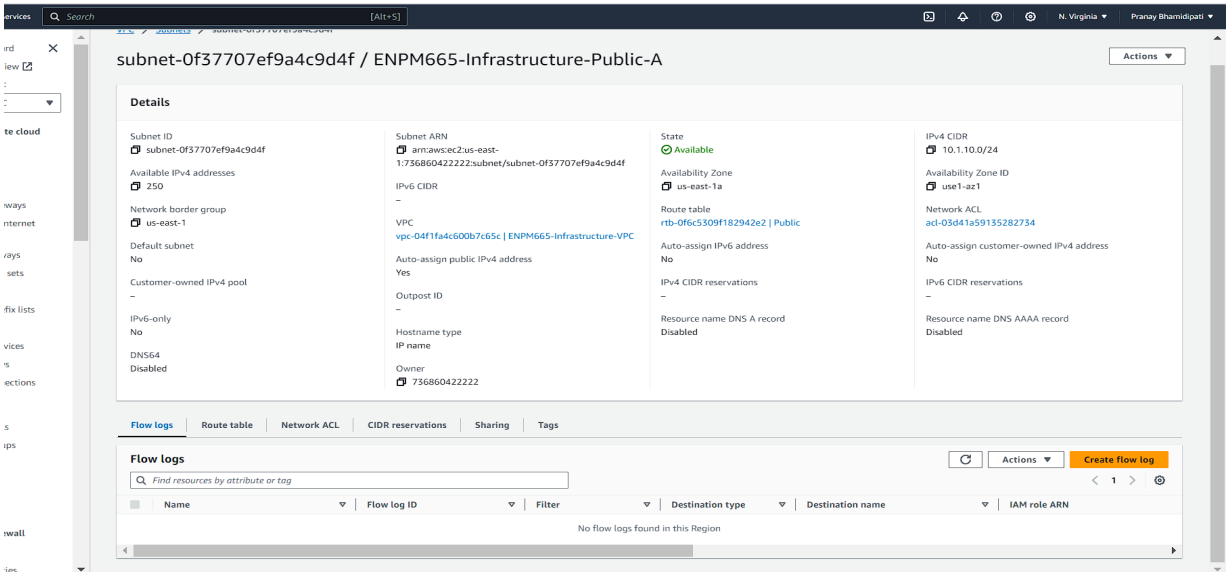
b. MedCircle should turn on VPC Flow Logs for their subnet of a particular VPC. Information like IP addresses, ports used, and more are recorded here. This applies to all traffic entering and leaving the company's VPC or subnet.

2. Medium severity vulnerabilities:

a. No Flow Logs in subnet.

Description:

Flow Logs are an AWS tool that if turned on will let MedCircle record various details about the data that comes into and goes out of the network. AWS Flow Logs give MedCircle useful information for fixing, protection, and following the NACL rules. (Workfall, Inc, 2023)



i. Vulnerability:

The absence of Flow Logs may result in the oversight of significant security incidents, like intrusion attempts, possible breaches, or the presence of harmful traffic.

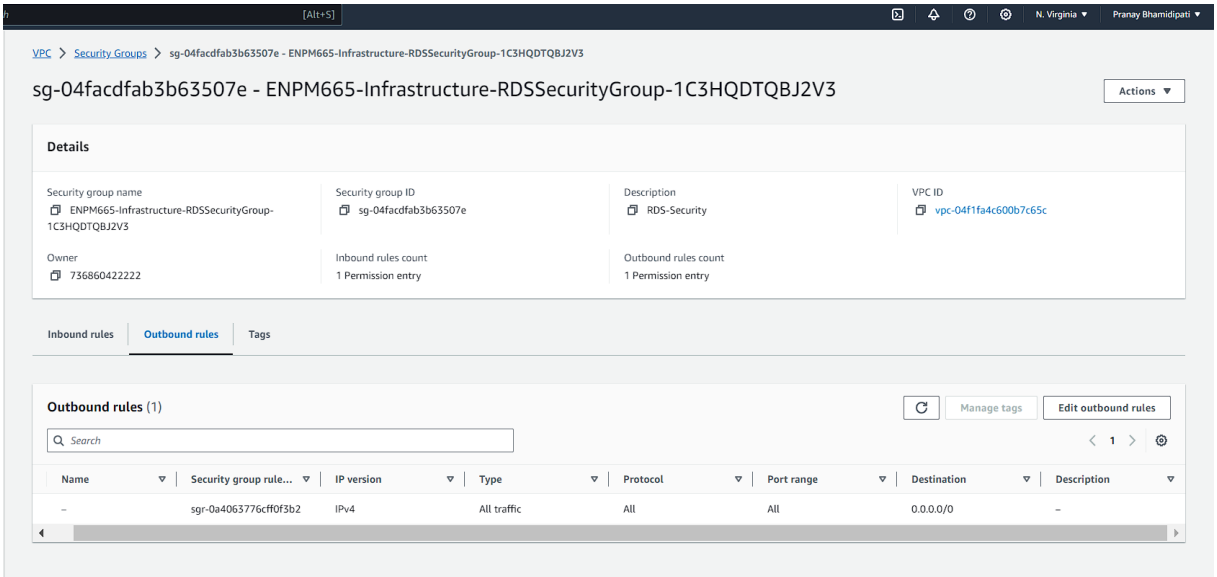
ii. Recommendation:

a. Ensure Flow Logs are enabled in a subnet and set up properly so that they record all of the pertinent traffic data and save it in a place for investigation purposes.

b. MedCircle should consistently evaluate and revise AWS Flow Log setups in order to effectively respond to modifications in the company's network.

b. Security group keeps all ports open:

Description:
The security group shows a lack of efficiency by failing to provide any kind of safeguard from unapproved access or harmful network traffic as all ports are open which can be exploited by hackers. (Teneble, 2023)



i. Vulnerability:

Exposing the company's resources to potential security threats, such as brute force attacks, Distributed Denial of Service (DDoS) attacks, and other security vulnerabilities, makes them susceptible to different forms of malicious intrusions.

ii. Recommendation:

When setting up permissions, security groups should always follow the concept of least privilege. Allow just the required ports to be used, and only from known, trustworthy sources.

V. Disaster recovery assessment report (Brush, 2022)

Executive Summary:

The cloud infrastructure of MedCircle is at substantial risk of data loss and extended downtime due to the absence of a comprehensive backup and disaster recovery plan. This report offers a more detailed analysis of disaster recovery readiness and proposes a comprehensive plan to mitigate the associated risks.

From the assessment

1. Absence of Backup Strategy:

   a. Severity: High

   b. Vulnerability:

      The infrastructure lacks a well-defined backup strategy, which includes the backup frequency, retention policies, and the types of data to be backed up. As a result, the company faces a high risk of data loss in the event of system failures, data corruption, or cyberattacks.

2. No Redundancy and Failover Plan:

   a. Severity: High

   b. Vulnerability:

      The infrastructure does not have redundancy or failover mechanisms in place. The absence of such mechanisms means that if any component or service experiences an interruption or failure, it could result in extended downtime and a loss of access to critical patient data. Redundancy and failover should be implemented for all vital components and services.

3. Inadequate Data Recovery Procedures:

   a. Severity: Medium

   b. Vulnerability:

      Although there might be partial backups of data, there is a lack of documented procedures for data recovery and system restoration. The company may face delays in data recovery and service restoration during a crisis, resulting in extended downtime.

4. Lack of Disaster Recovery Testing:

   a. Severity: Medium

   b. Vulnerability: The infrastructure has not undergone any disaster recovery testing or drills to evaluate its readiness and effectiveness in the case of an actual disaster. This lack of testing increases the uncertainty about whether the recovery procedures will function as intended.

5. Limited Backup Locations:

a. Severity: Medium

b. Vulnerability: The existing backup strategy stores all backups in the same data center or location. This approach poses a medium risk, as regional disasters or data center failures could lead to data loss. Implementing a multi-region strategy for backups is crucial to reduce this risk.

6. Insufficient Documentation:

a. Severity: Low

b. Vulnerability: The documentation related to the infrastructure does not include a comprehensive disaster recovery plan. The lack of clear and detailed procedures and documentation could hinder response efforts during a crisis. It is essential to have well-documented processes to ensure a swift and effective response.

Proposed Backup and Disaster Recovery Plan:

To mitigate the identified risks and enhance the disaster recovery readiness, a comprehensive plan is recommended:

1. Regular Automated Backups:

To implement automated daily backups for all critical data, including patient records, medical images, and databases. To define the backup frequency, retention policies, and the types of data to be backed up.

2. Data Redundancy:

To set up redundancies for critical components such as database servers, application servers, and network infrastructure in different availability zones or regions. To implement a failover plan to ensure uninterrupted service availability.

3. Testing and Validation:

Firm should conduct regular disaster recovery testing to ensure that the recovery procedures work as expected. This should include data recovery tests, system restoration tests, and full-scale disaster recovery drills.

4. Geographically Diverse Backups:

The firm should store backups in geographically diverse locations to safeguard against regional disasters. Utilize a multi-region strategy for backup storage where feasible.

5. Documentation:

Creating comprehensive documentation that outlines disaster recovery procedures, roles and responsibilities, communication protocols during a crisis, and step-by-step guides for recovery operations.

6. Monitoring and Alerting:

Implementing a robust monitoring and alerting system to detect incidents in real time. Setting up automated alerts to initiate the disaster recovery plan promptly when irregularities are detected.

7. Data Encryption:

Firm should ensure that backups are securely stored with encryption in transit and at rest. Implement encryption mechanisms to protect sensitive patient data from unauthorized access.

8. Personnel Training:

The firm should provide training to IT staff on disaster recovery procedures and best practices. Ensure that all relevant team members are well-prepared to respond effectively during a crisis.

9. Regular Review and Update:

The firm should establish a schedule for regular reviews and updates of the disaster recovery plan. Ensure that the plan remains up to date with the evolving infrastructure and changing threat landscapes.

Conclusion:

Medcircle's cloud infrastructure is at significant risk of data loss and extended downtime due to the absence of a comprehensive backup and disaster recovery plan. The proposed plan, when implemented and regularly tested, will mitigate these risks and ensure the continuity of healthcare services and the protection of sensitive patient data.

VI. References

Adkoli, J. H. (2018, September 21). *5 Best Practices for AWS NACLs (Network Access Control Lists)*. Retrieved from Dzone: https://dzone.com/articles/5-not-to-ignore-best-practices-for-aws-nacls-netwo

Amazon Web Services. (2023). *Security best practices for Amazon S3*. Retrieved from Amazon User Guide: https://docs.aws.amazon.com/AmazonS3/latest/userguide/security-best-practices.html

Amazon Web Services. (2023). *What is Amazon VPC*. Retrieved from Amazon Virtual Private Cloud User Guide: https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html

Brush, K. (2022, May). *disaster recovery plan (DRP)*. Retrieved from TechTarget: https://www.techtarget.com/searchdisasterrecovery/definition/disaster-recovery-plan

CloudYali Team. (2023, June 17). *Finding Unencrypted AWS EBS Volumes at Scale*. Retrieved from Cloud Yali: https://www.cloudyali.io/blogs/finding-unencrypted-aws-ebs-volumes-at-scale

Hacker News. (20233, July 24). *New OpenSSH Vulnerability Exposes Linux Systems to Remote Command Injection*. Retrieved from The Hacker News: https://thehackernews.com/2023/07/new-openssh-vulnerability-exposes-linux.html

Jyl, G. (2023, May 11). *AWS - VPC Security - NACL Network Access control list*. Retrieved from ocholuo.github.io: https://ocholuo.github.io/posts/NACL/

Lee, B. (2019, September 19). *Cloud Network Security 101: AWS Security Groups vs NACLs*. Retrieved from Cloud Security 101: https://www.fugue.co/blog/cloud-network-security-101-aws-security-groups-vs-nacls

Microsoft Inc. (2023). *What is identity and access management (IAM)?* Retrieved from Microsoft Security: https://www.microsoft.com/en-us/security/business/security-101/what-is-identity-access-management-iam

Teneble. (2023). *Tenable Cloud Secuirty Policyies*. Retrieved from teneble: https://www.tenable.com/policies/cloud-security/AC_AWS_0590

White, E. (2020, July 15). *Must-know best practices for Amazon EBS Encryption*. Retrieved from AWS Blog: https://aws.amazon.com/blogs/compute/must-know-best-practices-for-amazon-ebs-encryption/

Workfall, Inc. (2023). *How To Log, View And Analyze Network Traffic Flows Using VPC Flow Logs?* Retrieved from The Workfall Blog.