ANALYSIS OF AUTOMATED ZERO TRUST AWS HOME NETWORK FOR CONFIDENTIALITY AND AUTHENTICATION ISSUES

MSc Research Project
Cyber Security

Wahaj Rashid
Student ID: x23197960

School of Computing
National College of Ireland

Supervisor: Rohit Verma

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Wahaj Rashid ……. ………………………………………………………………………………………………

Student ID: X23197960 ………………………………………………………………………………………………..……

Programme: MSc Cyber Security Jan 2024-2025 ……………………………………………………………… **Year:** 2024-2025 …………………………..

Module: MSc Research Practicum II ………………………………………………………………………………………….………

Supervisor: Rohit Verma ………………………………………………………………………………………….………

Submission Due Date: 12-12-2024 @ 02:00pm ………………………………………………………………………………………….………

Project Title: ANALYSIS OF AUTOMATED ZERO TRUST AWS HOME NETWORK FOR CONFIDENTIALITY AND AUTHENTICATION ISSUES ………………………………………………………………………………………….………

Word Count: 8221 …………………………………… **Page Count** 19 ………………………………………………..……..

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……………………………………………………………………………………………

Date: 12-12-2024 ………………………………………………………………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on a computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

ANALYSIS OF AUTOMATED ZERO TRUST AWS HOME NETWORK FOR CONFIDENTIALITY AND AUTHENTICATION ISSUES

Wahaj Rashid

X23197960

Abstract

In a home network environment, this research investigates the issues of maintaining confidentiality and ensuring authentication issues that involve unauthorized access, data leakage, and insufficient segmentation of the network which are common. Moreover, this research also addresses traditional approaches like Virtual Private Network-based solutions which often fail to effectively address the evolving challenges of securing home networks, particularly when it comes to managing access for multiple users and devices while maintaining high levels of confidentiality and authentication. These issues could be resolved by using a Zero Trust orchestrator like Twingate and its automation with Infrastructure as Code could help to integrate the architecture in a home network for a secure environment. This research explores and evaluates the principle of Zero Trust which is “never trust, always verify” to prevent security risks that involve confidentiality and authentication with automation. It shows how the Zero Trust framework plays its role in home security including methods of Multi-Factor Authentication, Access Control, Biometrics, Encryption mechanisms like AES256 and KMS, and network segmentation to prevent data leakage and ensure confidentiality. Moreover, it allows verified and authenticated users from outside the network to get into the network and use its resources, which could not be possible by using a Virtual Private Network. To justify the Zero Trust Model in this research. A Cloud platform AWS has used in the network with its services like for computing EC2 and S3 for encrypted storage and Access Control List for policy enforcement. These services are tested to provide a secure environment for a user in a home network and its whole configuration is done by using Infrastructure as Code that automates everything and provides a scalable, repeatable Zero Trust model with fewer chances of human error involvement and time-saving The results in the form data encryption, network segmentation and access management with user and its device authentication validates that Zero Trust in resolving the security issues critical in a home network and provide resilient environment that could be integrated with automation instead of using traditional VPN solutions. This research not only goes through the Zero Trust principles but also validates its automation that can be used for both enterprise and personal networks to overcome the challenges of confidentiality and authentication.

1 Introduction

Due to increasing reliance on cloud services, home network security becomes an essential component these days because there is a rise in cyber threats in a home network if there is a use of the cloud and its resources inside the network (‘Could Greater Reliance on Cloud Computing Mean a Cyber Security Risk…’ 2023). Some home network use Virtual Private Network - VPN to connect the devices in the same network but it is an old and traditional way and does not include an extra layer of security the data that they are storing and retrieving from the resources are not strongly encrypted and comes with the chance of data leakage that raises the issues of confidentiality and authentication in a home network (Gwilliam, 2022). Network security models like Dmz-based architectures VPN-Based Access Network Access Control (NAC) with Static Rules and Castle-and-Moat Model are old and rely on perimeter defense base mechanism which is not good enough to identify threats inside the network, restrict unauthorized access, proper network segmentation, and strong data encryption (VPN Replacement | Docs 2024). To resolve these confidentiality and authentication issues Zero

Trust model with AWS resources in a home network came into place with the principle of "never trust but always verify" and this research evaluates the effectiveness of Automation in the Zero Trust Model with AWS services in a Home Network to resolve these issues with authentication, confidentiality, and access management by deploying strong encryption, network segmentation, user access management with multi-layer authentication and biometric (Sarkar et al., 2022). The goal and motive of this research are to present by evaluate the effectiveness of this approach for a home network that not only offers adaptability and scalability but also resolves the issues in confidentiality and authentication for a home network.

1.1 Research Question

How confidentiality and authentication issues can be automated in Zero Trust AWS for the home network environment?

1.2 Research Objectives

The objectives of this research include:

1. To develop and design a secure home network framework based on Zero Trust principles for all the verified users and devices that ensure robust confidentiality and authentication and address its related issues.
2. To evaluate the effectiveness of Zero Trust confidentiality and authentication integrate restricted and secure access with an Access Control List for users and Multi-Factor Authentication for devices to access the network resources and implement the encryption techniques for data transmission security instead of using the default encryption mechanism.
3. To eliminate the process of manual configuration with the integration of an automated framework (Infrastructure as Code) that configures and deploys the network policies with its resources and reduces the ratio of human error.
4. To determine whether the given research is preventing security threats related to confidentiality and authentication including network segmentation issues, unauthorized access, and data breaches or not.

1.3 Contribution to the Scientific Literature

Zero trust in a home network was already used by many researchers but there were some authentication and confidentiality issues in it because without the usage of VPN (The future of VPN: transitioning from traditional VPNs to zero trust, n.d) and not using Automation for Zero Trust Principle. Zero Trust managing console which is Twingate and acts as an alternative VPN resolves these issues with different policies of AWS and provides secure access management the whole project is automated by using Terraform which is Infrastructure as code and automates the whole configuration and deployment steps and reduces the chances of human error and save time to configure the whole home network manually, the research demonstrates a practical approach to achieving scalable, error-resistant confidentiality and authentication mechanisms and resolve its issues by deploying strong encryption, network segmentation, user access management with multi-layer authentication and biometric. This research bridges the gap between theoretical Zero Trust models and their practical application in home networks with automation by evaluating its effectiveness to overcome the challenges of authentication and confidentiality which was not done by the past research (Wang et al. 2024).

1.4 Structure of the Report

The remainder of the report is structured as follows. In section 2 Literature Review will be discussed including the related work done by the researcher with its gap. Meanwhile, Section 3 will discuss the Methodology including Implementation and experimental setup with Evaluation metrics and Criteria for the thesis, and Section 4 will discuss The Design Specification of the research and its main elements. Section 5 is based on the Implementation if the Zero Trust architecture with its resources Section 6 is the Evaluation part and Section 7 Conclusion and Future Work will be discussed, and Lastly, Section 8 is for the references.

2. Literature Review

This section gave analysis and discussion on the existing literature on Zero trust model with confidentiality and authentication in cloud like AWS and it identified the gaps that could be filled by this research paper. Every subsection discussed the contribution, limitation, gaps, and problems that could be addressed by this research.

2.1. Why Zero Trust Architecture Should Be Considered?

At the beginning companies and home network used to prefer network models like “the castle and moat” which is perimeter security to build a secure boundary for their network and they used Intrusion detection system, Intrusion prevention system and firewalls (‘ZTA’ 2023). This approach works like once the device and user are authenticated, they will not be authenticating again, and they are trustworthy, and this raises the amount of insider threats vulnerability including reliance on cloud services and attacks that bypass security measures (Omier 2022). Against these vulnerabilities zero trust comes in action under the principle of “never trust, always verify” which means no one is trustworthy even if they are from inside the network they need to go through the authentication and authorization phase to access the network resources. It focuses on strong authentication, regular access control, continuous monitoring, least privilege principle and micro-segmentation and these elements are not available for a user using traditional techniques in a network. In 2012 John Kindervag introduced Zero Trust in Palo Alto Network after that numerous numbers of studies have been conducted on whether it is for small home network or big industrial networks (‘Build Security into Your Network’s DNA’ 2013). A guide on Zero Trust Architecture has been published by the National Institute of Standards and Technologies which provides organizations with a framework to analyze this architecture to deploy and design the solutions of Zero Trust. (Weinberg and Cohen 2024) In 2020 NIST also released “NIST SPECIAL PUBLICATION 1800-35E: Implementing a Zero Trust Architecture” which comes with the in-depth ZTA implementation exploration and how it could be used in Practical applications and reduce the chances of errors and enhance confidentiality and authentication and provide a detail guide how to move from traditional perimeter security model to adaptive and robust Zero Trust one (Implementing a Zero Trust Architecture 2023). This identified gap was that the organization or the user must analyze each step manually and there was no automation for configuration, policies and deployment was going on and that problem could be resolve by considering a service Infrastructure as Code and Zero Trust Managing Console that was used in this research to enhance the security and reduced the human error while configuration was done in minimum amount of time.

2.2. Zero Trust with Cloud Platforms and an Alternate For VPN

Researchers have studied in recent years the Zero Trust model on different IT technology like cloud and Internet of Things. The challenges and best practices Related to the implementation of Zero Trust for remote access was investigated M. K. Pratt on March 2023, and the researcher provided the steps and insights to move from VPN to Zero Trust based on remote access. This research provided the integration steps, but the limitation that they were having financial constraints, using legacy technologies, increased friction makes user pushback, and faces complexity. Only 2% achieved stages like password-less access, researcher highlights that there was a need of better techniques to improve the implementation process with confidentiality and authentication for users and devices. This thesis is resolving the issues of implementation because it uses twingate which provide access management and user access from the admin side and deployment and configuration with its implementation is easy to understand and the resources are available on their official sites ('What Is Zero Trust?' 2023). The research on securing remote access to educational online laboratories using Zero Trust Network Access (ZTNA) shares a conceptual link with this study, particularly in the application of Zero Trust principles to secure environments. Both works aim to address security risks and unauthorized access, albeit in different contexts: one focuses on online laboratories in educational settings, while the other focuses on home networks. Remote access to the online labs with security protocols was one of the gaps that are required to be performed by the future researchers. This research, on the other hand, evaluated automation of Zero Trust within a home network environment using AWS resources, addressed the specific challenges of confidentiality and authentication within a more private setting. The automation of device and user verification, multi-layer authentication, and encryption through KMS in this research validated a further enhancement to the principles outlined in the related work, focusing on the technical execution and security of home networks rather than a purely conceptual or manual approach (Tuyishime et al. 2024).

2.3. Confidentiality Issues with Cloud

At the beginning of the AWS a security analysis was conducted by Modi el al and the researcher find various security issues like there is a need of data security for the services and resources in AWS wheatear data is at transit stage or rest also there should be improved access control mechanism and identity management. Most of the issues till now are underscored the importance of its security adoption in a network and the services used by the users (Modi et al. 2013). This paper surveys different intrusions that affected the performance of the cloud with its confidentiality and the future work was described as using better tools and technology for security should resolve these issues. Now most of the issues are resolved by using different AWS services like IAM roles, KMS and Cloud Watch etc. but these tools need to be configured manually, and not all the services have default access of these tools. In this research S3 as a resource for user has integrated with the data encryption at state of rest and transit. S3 comes with its default encryption mechanism but to enhance it further KMS and logging mechanism has integrated to provide extra layer for confidentiality.

2.4. Mitigate Inside Threat with Authentication

As the technology enhanced and users started moving to cloud due to its benefits the threats that comes with it are also increased by the time and inside threat was one of its types that included compromised credentials, malicious intent, loss of trust, sensitive data leakage and reputational damage. To overcome those type of threats the implementation of Zero Trust with Role Based Access was advised because that not only ensured the user access according to their assigned roles but also integrated the least privilege policy for the users in the network. Moreover, the triggering alerts, activity logs and other anemology detection techniques helped researchers to monitor and prevent anomalies (Ahmadi 2024). It relates to this research because this research was based on the principle of Zero Trust and the confidentiality and authentication challenges was evaluated by testing the user access, encryption of the data and ensured only verified and authenticated account got access in the network. Issues like data leakage, unwanted access from unauthorized users, exposure of resources in public, human error, and configuration complexity with downtime in deployment were resolved with the automated model of Zero Trust (Siddhartha et al. 2023).

2.5. Integration of Automated Zero Trust network

An autonomic security system that aligned Zero Trust principles and prevented against threats from both outside and inside the network, with the ability to integrated with cloud-based platforms was introduced in 2017 that resolve various issues, but the researcher used traditional methods and techniques with different frameworks which is complex to analyze especially for a home network user. On the other side this thesis is integrated with twingate that act as a bridge between client and the resources through its connector and provide flexibility and ease of adaptability to integrate manually and automate it using terraform and numerous numbers of resources could be integrated and automated just by simple understanding of policies (Eidle et al. 2017). The case study by Wang integrated Transparent shaping and ZTA which is Zero Trust Architecture into an application which was hosted by AWS and maintain the functionality with the enhancement of security was demonstrated. This aligns with the focus of this thesis on resolving confidentiality and authentication issues in a home network using automated Zero Trust principles in AWS. While the referenced study highlights manual implementation and suggests automation as future work (Wang et al. 2024), this thesis addresses this gap by automating ZTA through Terraform, ensuring scalability and efficiency.

3. Methodology

This is experimental based research with qualitative and quantitative research methodology with a comprehensive approach to understand How Confidentiality and Authentication Issues Can Be Automated in A Zero Trust Aws for Home Network Environment and the motive is to research the effectiveness of using it.

3.1 Overview

The methodology of this research involves a Zero Trust framework to address authentication and confidentiality challenges in a home network with principle of Zero Trust which is “never trust, always verify” to prevent from security risks that involves confidentiality and authentication. This approach shows how Zero Trust framework plays it role in home security by including methods of Multi Factor Authentication, Access Control, Biometrics, Encryption mechanisms like AES256 and KMS and network segmentation to prevent data leakage and ensure confidentiality with authentication. The network components, configurations and security policies are automated through Infrastructure as Code (IaC), ensuring consistent and scalable deployment also reduce human error ratio and eliminate manual configuration. Zero trust console has been utilized to manage secure access and manage users in the network and that eliminates the need of VPN by overcoming the issues related to confidentiality and authentication.

3.1. Implementation of Zero-Trust

Its implementation includes an account on AWS and Twingate and to automate the whole project with configuration of AWS and Twingate also used Terraform which is infrastructure as code. Before deploying the code, there is need to make sure that this project has configured terraform and AWS cli in our local machine and AWS cli should connected with the AWS root account. After deployment Terraform, configuration automates the deployment of a Zero Trust environment in AWS, utilizing Twingate to enforce confidentiality and authentication across a home network setup. A (VPC) Virtual Private Cloud with both public and private subnets contained by CIDR block, give access for controlled access, and segregated to resources (Vundavalli 2024). Within this environment, multiple EC2 instances are created, including a dedicated Twingate connector instance that authenticates access to private resources through Twingate's secure tunneling. This project would have two S3 buckets in our public EC2, one for storing users' data and other for generating and storing logs to monitor the activity of the user in S3. In terms of **confidentiality** this structure provides data protection in resources like S3 with EC2 instance also it follows strict access control and transparent mechanism. The S3 bucket files can only be used or accessed by authenticated users by enforcing HTTPs access only due to bucket policies that integrated in the terraform code and it also prevents insecure data transfer. ACL defines which account, or group can access the files in the buckets, and it is based on policies and access groups. S3 bucket and its files are by default encrypted with SHA256, but this research provide additional layer of encryption using KMS (AWS Key Management Service - AWS Key Management Service 2024). Additionally, there is a separate bucket that will be generated after the code deployment, and it will only store logs and activities happening in our main bucket. Every single resource including public EC2 will be hidden from unauthorized users and only will be viable to those users or a group of users that has been defined in the code. In terms of **authentication** after deployment user will receive an invite in their emails and logged in Twingate client with same email and after that the Twingate client application authorize user by navigating it to MFA step and also allow user to set a password less login like biometrics to improve theory usability when login is successfully it will let user to see the IP addresses of the available resources and user will get access to those resources by using a third part terminal if they are using a mobile device (Two-Factor Authentication | Docs 2024). The successful and failed authentication rate is depend on the device and the account that the user is using if admin knows the account and the device and verified it only then user can access those resources otherwise it won't be accessible and user will get blocked and this is how it is improving the authentication issues because when a user use VPN then there is no such concepts of MFA or verified devices and accounts so that is how Zero Trust is resolving the issue of authentication in it by applying multiple layer of protection for authentication. Services from AWS like IAM and policies from Twingate ensure that only resources give their access to verified users and authenticated devices and that creates a barrier Infront of unauthorized users to access those resources (Device Security Controls | Docs 2024).

3.2. Experimental Setup

- Environment Configuration:
    1. For deployment, this research is using a Windows 11 system as such there is no requirement of high-level specification but to setup the environment user must have VS code with latest npm version to install the packages and user must have latest version of terraform and AWS cli (AWS cli must be connected to the AWS root account). This configuration part for each tool can be done by their official documents.
    2. To connect the devices and use the AWS resources in a home network each user should have an application which is Twingate client which enables the connectivity with network to the user and to run those resources user should use some SSH terminal this project is using Termius.

3. Twingate integration is important because to manage the connected devices and monitor them this project is using Twingate admin that allows to monitor and ensure that only authenticated devices can access AWS resources.

- Deployment of AWS resources using Terraform, including EC2 instances for running applications and S3 buckets for user storage and Twingate connectors for network access management which provides the connectivity with the resources and network to the connected devise which are using twingate client.
- Resources: EC2 and S3 are configured in the terraform main.ts file and after deployment user with privileges can use it according to their roles that are assigned in the terraform code with twingate.

3.3. Evaluation Metrices

- Access Logs: A dedicated S3 bucket for logs was generated to store performing activity in the S3 bucket and the activity of the device and resources were stored in Zero Trust web-based admin console which is Twingate Admin.
- Performance Metrics: Evaluated the performance of EC2 and the response times of confidentiality and authentication processes include time of deployment of the Zero Trust architecture.
- Failure Rates: Evaluated the success rate to establish a connection with the network and its resources.
- Encryption Status: Assessed using AWS Config to ensure all data is encrypted in transit and at rest.
- Access Control List Compliance: Tested the restriction of ACL for public users.
- Authentication Metrics: Measured the Adaption rate of MFA for the users in the network.

3.4. Evaluation Criteria

This thesis is being **evaluated** by the resolution of authentication and confidentiality issues in a Zero Trust AWS home network environment using automation. To assess confidentiality, there is analyzed encryption and decryption times for S3 buckets integrated with AWS KMS, using Python scripts to quantify the performance impact during tasks like storing large files. To analyze and capture logs activity from the main S3 bucket a dedicated S3 bucket for logs has been configured in AWS, ensuring traceability and secure storage. For authentication, there are monitored failed and successful access attempts using the Twingate admin dashboard, categorizing access patterns to evaluate the effectiveness of Zero Trust policies. Additionally, the CPU utilization of both public and private EC2 instances during operations like file transfers is measured to understand the performance impact of the analyzed security measures. All evaluations were automated using Python scripts, demonstrating the practicality and scalability of automation in resolving security issues within a Zero Trust framework.

4. Design Specification

The design is focused on providing a Zero Trust Environment in a home network with a few numbers of devices since it is a home network for home users so there is no need to connect many devices in the network. In Figure. 1 Devices in the network relate to AWS resources that provide its services to the connected devices and those services and resources are only accessible and visible to the device with the overall access of the network which means only for trusted and verified devices. For example, if an Android device stores some file in an S3 bucket then it could be visible to the IOS device because each of them is connected to a single network.

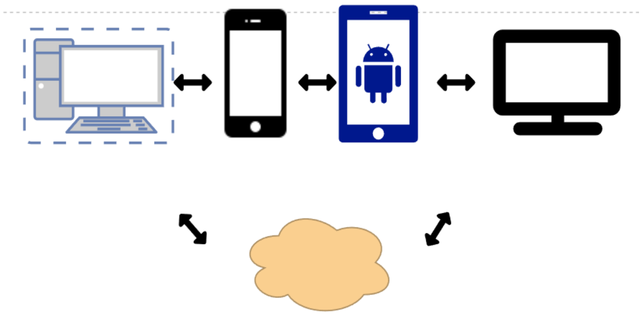


Figure 1: Home Network connected to multiple devices and using AWS resources

Zero Trust comes with the principle of "never trust always verify" so the research designed it with multiple layers of authentications with confidentiality. To perform the highest level of confidentiality and enable zero trust in the network by using Twingate that is an orchestration to provide secured access. To reduce human error and reduce the time of manual configuration to create the whole setup by using Terraform which Infrastructure as code and gives a scalable and agile solution for the network.

4.1. Home network integration with AWS cloud services

After the deployment of terraform code, it will create a Virtual Private Cloud with several private IPs in it after that it contain public and private subnets with a smaller number of IP's as compared to our VPC and the private subnet is hosting EC2 type of resources (Aws resources and shared VPC subnets - amazon virtual private cloud, n.d.). These resources contain private IPs, and they are not accessible to the public Internet directly. After that there is an association with our subnet to the route table to manage the flow of the traffic to make sure only verified and authenticated devices can request access to the resources. Not only could the users in the home network or family members access those resources but also the external user could access those resources too as shown in Figure. 2, but that user needs to be authenticated and verified by the admin of the network first.

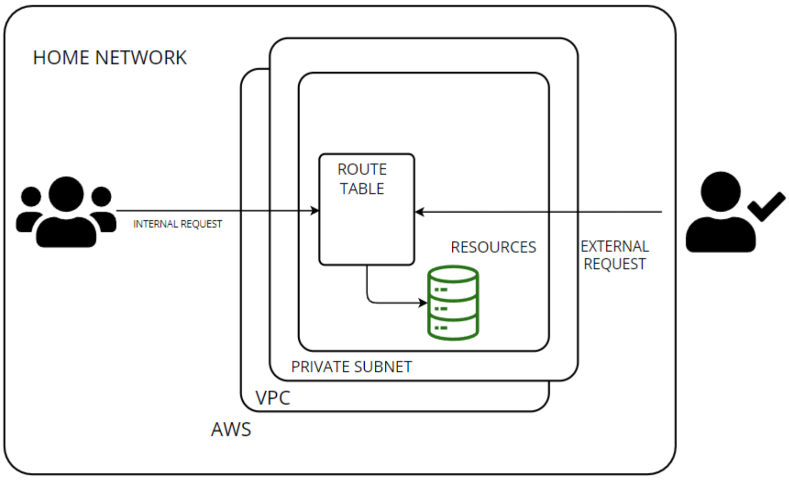


Figure 2: Architecture of Home Network with Zero Trust using AWS

4.2. Twingate Admin Dashboard

Twingate admin dashboard can manage all the devices and montoring them but mainly it is used to give access to the users and their devices to access the resources that they are suppose to have. Once the deplyment is done Twingate admin will see the devices and users on the screen and the user need to get verified it self and its account first by the Twingate Admin to access the AWS resources. As shown in Figure. 3 If user device and its account have access then the resources could be accessed and if the device is not veriified by the admin even if the user account have access then the reources will not ne accessible by the user and if user account and its devices are not valifdated by the admin then the user will get blocked and could not access the resources.

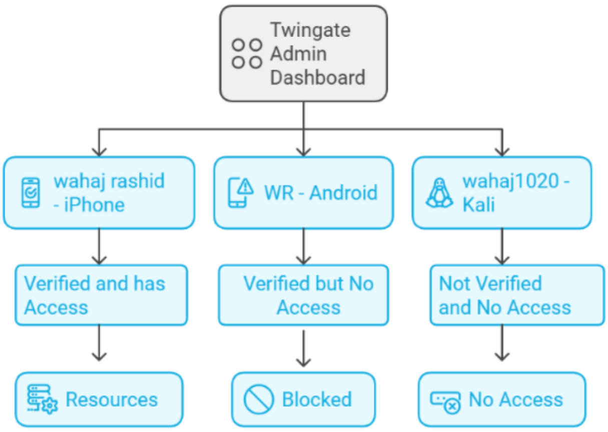


Figure 3: Scenarios of Accessible and Non-Accessible Devices for the Home Network

4.3. Twingate Functionality And Structure

Twingate is built to replace old VPS by providing remote access, cloud infrastructurem, private remote access and additional layers of security and solution for security in a zero trust. It could be integrated with latest tools and simple setups for any type of user. Its important features include DNS filtering, access management for infrastructure, comptibilty with different platforms and devices and ensure solutions that are efficient, scalable and secure for users in home network and intustrial users like DevOps teams (How to Use Terraform with AWS and Twingate | Docs 2024). As shown in Figure. 4 it allows devices to connect the AWS resources like EC2 and S3 bucket by providing a secure connection between them without exposing the network. Groups are used to define policies which are kind of rules and restrictions. Resources section in twingate reresent the resources that this research is using from cloud platforms like in AWS this research using EC2 and S3 and Connectors built a secure and encrypted connection to the home network which is private . Within the VPC in AWS these connectors are being configured and deployed so specific subnet could be operated which allows AWS resources a secure interface. Through the policies and twingate network when system allows devicees and users to access the resources then the connector works like a bridge between the AWS environment and client and gives secure access to the resources like EC2 instances without publicizing the subnet or exposing the VPC by following the principles of to Zero Trust.

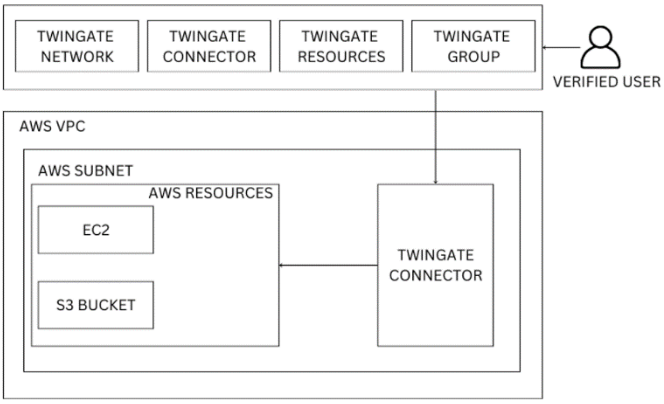


Figure 4: Architecture of Twingate with AWS

4.4. Terraform Deployment

This research used Terraform which is infrastructure as code. Terraform configuration automates the deployment of a Zero Trust environment in AWS as shown in the Figure. 5, utilizing Twingate to enforce confidentiality and authentication across a home network setup. A (VPC) Virtual Private Cloud with both public and private subnets contained by CIDR block, give access for controlled access, and segregated to resources. Within this environment, multiple EC2 instances are created, including a dedicated Twingate connector instance that authenticates access to private resources through Twingate's secure tunneling. Connection of resources is relied on the defined groups and resources in the policies of Twingate and that provides a layer of authentication of resources by restricting them from unauthorized users for a specific resource in the network. This research has created multiple buckets one for storage and other for generating and storing logs to monitor it and the data which is being transferred from EC2 to S3 bucket was encrypted by default, but this research added KMS to add additional layer of encryption (ZeroKMS: Zero trust key management, n.d.). Once's it is successfully deployed the resources will automatically be visible to twingate admin and a request to access the resources to authenticated users will be sent through email and once they accept the request, they will be able to access those resources if their devices and IOS environment are authenticated by the Twingate Admin too.

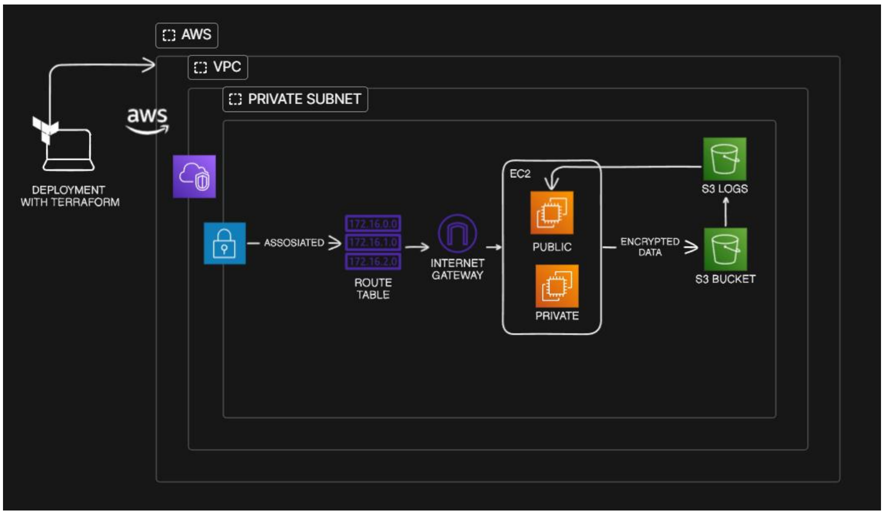


Figure 5: Architecture of code deployment using Terraform with AWS

4.5. Authentication Techniques

There are multilayer of authentication has used since the main focus is to enhance the authentication in a zero trust environment so this project analyzed Access control list for S3 bucket for specific user, integrated MFA for twingate client and AWS resources also user can setup password less authentication for its ease of use and the encryption part data is going to be encrypted by default because of the configured policies and twingate mechanism and the data traveling to S3 from EC2 is further improved in encryption by implementing KMS in it.

5. Implementation

Since the research is focused on the improvement and resolving the authentication and confidentiality issues in zero trust home network using AWS which will also reduce the human error and provide and string protection against unauthorized access and threats. The whole design has described in the previous section number 4 and the results from the thesis code is two EC2 instances public and private and a S3 buckets which can be used by our public EC2 even the public EC2 have public Ip, but it is not going to be visible or accessible to unauthorized users and unverified devices. The deployment part is the final part of the project because it automates the whole configuration for AWS EC2 and S3 with twingate. So, the output this thesis get is configured EC2 public and private and S3 bucket both are having policies that is configured in the code already and providing addition layer of security like not being accessible for HTTP or giving internet connection and for S3 having default encryption setup by S3, but this research further enhanced it using Key Management Service to give and additional layer of encryption to enhance the confidentiality for the data in our Zero Trust Environment. The terraform approach of Infrastructure as Code resolves the issue of automation reduce human error and with minimum efforts for manual and repeated configuration it reduces time too and make the whole setup fast, easy and reduce the level of failures and human errors. The code written in TF file follows the ACL language, which is easy to understand, and it provides defined segmentations of the network, enforcement of strict access rules with user access level and security groups within the home network. To access the resources in a home network twingate client app should be installed in the mobile devices and in kali, Mac or IOS system too. If the user gets the request to connect its device through the email, then this client application will make resources visible to the users with their respected IP address and once user have the IP address then user can get connected to those resources with any SSH application if user is using mobile device Termius APP is recommended. Every time user is trying to get into the home network for the first time then user must go through the MFA process and must have an authenticator app after that user can set password less authentication in the form of Biometric like fingerprint or face unlock. Moreover, policies and role-based access can be managed by the Twingate Admin and could be define in code for S3 bucket and its files using ACL

5.1. Security Enhancements:

- The segmented network isolates sensitive resources and reduces the chances of lateral movement performed by unauthorized users.
- Secure handling of user files prevents data leakage, using different encryption types.
- Auditing and monitoring have automated logs generation that helps to identify unauthorized activities within the network and to mitigate the vulnerabilities.
- **Key Outcomes:**

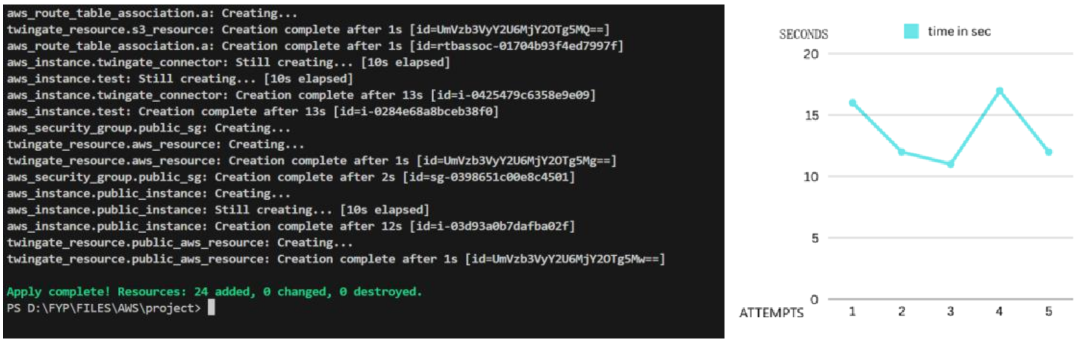
- Automated zero trust home network that does not require any manual configuration reduces chances of human errors and saves manual configuration time.
- Enhanced encryption from default AES256 to KMS for data in S3 bucket.
- Improved authentication by integrating multiple layers of security and allowing user to use password less biometric option while logging.
- Based on administrative policies allowing internal and external users to get into the network by providing a flexible and secure infrastructure.

6.0. Evaluation

The main finding, experiments and the research will be discussed in this section by addressing the research question "How confidentiality and authentication issues can be automated in a Zero Trust AWS for a home network environment?".

6.1. Experiment 01 From Deployment Site

The automation is done by terraform code in which we configured AWS structure with its services like EC2 and S3 and configured twingate with its user groups, connectors, resources, and policies also for user we mentioned the email of the users of our home network so they could get the invite after deployment. After the deployment there could not be any error because there was no involvement of humans while configuration. But still to identify errors before deployment, we ran command on terminal “terraform validate” and that command checked all the test and succeeded from the configuration site. The deployment did not take more than 20 seconds in 5 numbers of attempts as shown in the Figure. 6.



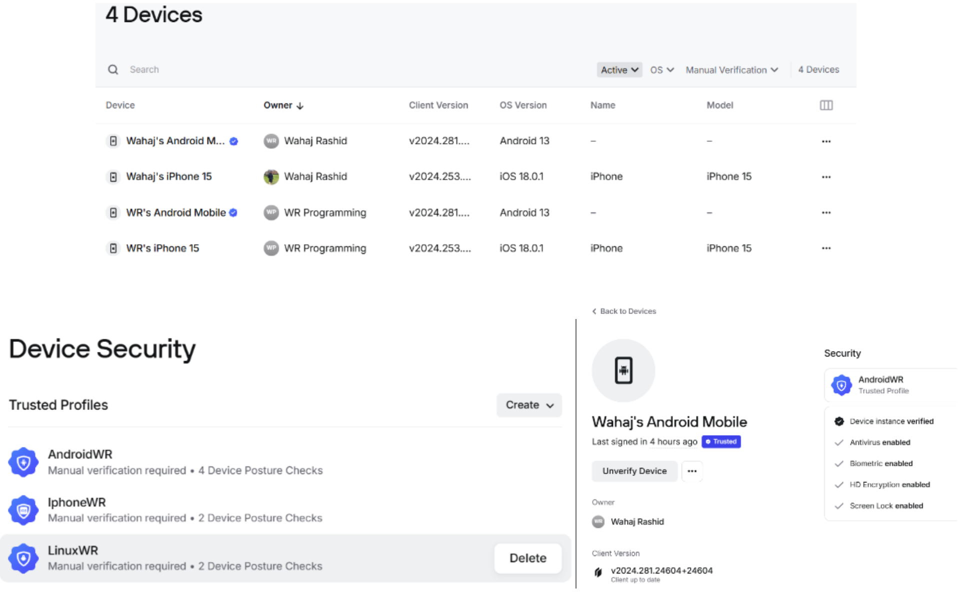
```
aws_route_table_association.a: Creating...
twingate_resource.s3_resource: Creation complete after 1s [id=UmVzb3VyY2U6MjY2OTg5MQ==]
aws_route_table_association.a: Creation complete after 1s [id=rtbassoc-01704b93f4ed7997f]
aws_instance.twingate_connector: Still creating... [10s elapsed]
aws_instance.test: Still creating... [10s elapsed]
aws_instance.twingate_connector: Creation complete after 13s [id=i-0425479c6358e9e09]
aws_instance.test: Creation complete after 13s [id=i-0284e68a8bceb38f0]
aws_security_group.public_sg: Creating...
twingate_resource.aws_resource: Creating...
twingate_resource.aws_resource: Creation complete after 1s [id=UmVzb3VyY2U6MjY2OTg5Mg==]
aws_security_group.public_sg: Creation complete after 2s [id=sg-0398651c00e8c4501]
aws_instance.public_instance: Creating...
aws_instance.public_instance: Still creating... [10s elapsed]
aws_instance.public_instance: Creation complete after 12s [id=i-03d93a0b7dafba02f]
twingate_resource.public_aws_resource: Creating...
twingate_resource.public_aws_resource: Creation complete after 1s [id=UmVzb3VyY2U6MjY2OTg5Mw==]

Apply complete! Resources: 24 added, 0 changed, 0 destroyed.
PS D:\FYP\FILES\AWS\project>
```

Figure 6. Time Taken to Deploy the Project

To resolve the issues of confidentiality and authentication with automation cannot be done by using VPN that is why this research integrated twingate and utilized it in terraform code with AWS resources and automate whole process so once the code is deployed it is available for these resources who are authenticated with verified devices. Twingate Admin or Home Network Admin is responsible for allowing access for devices, user account and environment of their device OS.

6.2. Experiment 02 From Admin Site

After establishing successful Zero Trust model our Twingate admin dashboard was able to manage all the devices and montor them but mainly it was used to give access to the users and their devices to access the resources that they are supposed to have. Once the deployment is done for Twingate admin console all the devices and users on the screen were visible and the user required to got verified it self and its account first through Twingate Admin to use the AWS resources. If user device and its account had access then the resources could be accessed and if the device was not veriified by the admin even if the user account had access then the reources would not ne accessible by the user and if user account and its devices were not validated by the admin then the user would got blocked and could not access the resources. That is the reasin why the admin is reponsible to maintain the principles of Zero Trust and user access management. As showin in the Figure. 7 device with bluc mark are only have access to the resources.



4 Devices

Search | Active | OS | Manual Verification | 4 Devices

Device	Owner ↓	Client Version	OS Version	Name	Model
Wahaj's Android M... ✓	Wahaj Rashid	v2024.281....	Android 13	–	–
Wahaj's iPhone 15	Wahaj Rashid	v2024.253....	iOS 18.0.1	iPhone	iPhone 15
WR's Android Mobile ✓	WR Programming	v2024.281....	Android 13	–	–
WR's iPhone 15	WR Programming	v2024.253....	iOS 18.0.1	iPhone	iPhone 15

Device Security

Trusted Profiles

AndroidWR
Manual verification required • 4 Device Posture Checks

IphoneWR
Manual verification required • 2 Device Posture Checks

LinuxWR
Manual verification required • 2 Device Posture Checks

< Back to Devices

Wahaj's Android Mobile
Last signed in 4 hours ago Trusted

Owner
Wahaj Rashid

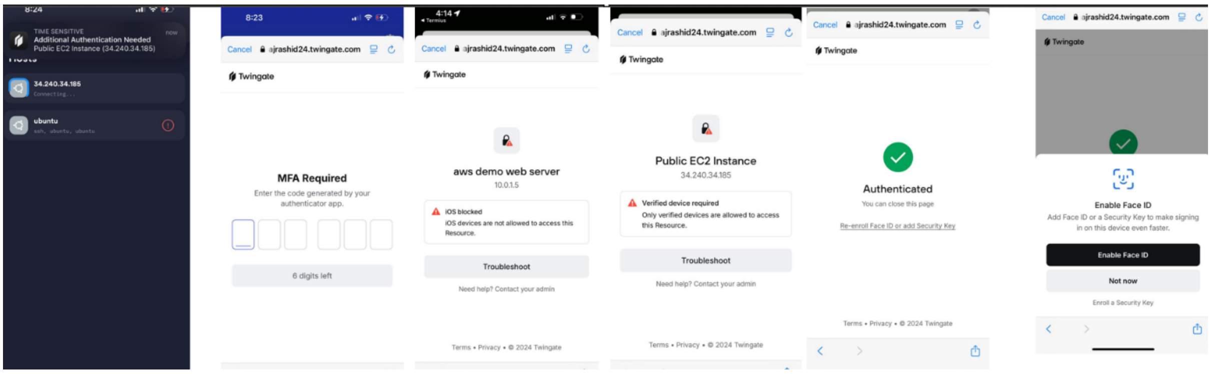
Client Version
v2024.281.24604+24604
Client up to date

Security
AndroidWR
Trusted Profile

- Device instance verified
- ✓ Antivirus enabled
- ✓ Biometric enabled
- ✓ HD Encryption enabled
- ✓ Screen Lock enabled

Figure 7 Devices in the network under Admin Control– Admin Panel View

6.3. Experiment 03 From Client and Authentication Site



TIME SENSITIVE now
Additional Authentication Needed
Public EC2 Instance (34.240.34.185)

34.240.34.185
Connecting...

ubuntu
ssh, ubuntu, ubuntu

MFA Required
Enter the code generated by your authenticator app.

6 digits left

aws demo web server
10.0.1.5

iOS blocked
iOS devices are not allowed to access this Resource.

Need help? Contact your admin

Terms • Privacy • © 2024 Twingate

Public EC2 Instance
34.240.34.185

Verified device required
Only verified devices are allowed to access this Resource.

Need help? Contact your admin

Terms • Privacy • © 2024 Twingate

Authenticated
You can close this page

Re-enroll Face ID or add Security Key

Terms • Privacy • © 2024 Twingate

Enable Face ID
Add Face ID or a Security Key to make signing in on this device even faster.

Figure 8. Client Site Login Phases

As shown in the above Figure. 8 The client site involvement was based on having a twingate client and a SSH terminal on their devices if they were using phones or computers. The user needed to accept the invite from the admin and after that the twingate client application authorized user by navigating it to MFA step and allow user to set a password less login to improve their usability when login was successfully done then it let user saw the IP addresses of the available resources and user got access to those resources by using a third-party terminal if they were using a mobile device. The successful and failed authentication rate was depend on the device and the account that the user was using if admin knew the account and the device and verified it only then user can access those resources otherwise it would not be accessible and user would get blocked and that is how it is improving the authentication issues because when there was a use of VPN then there was no such concepts of MFA or verified devices and accounts so that is how Zero Trust is resolving the issue of authentication in it by applying multiple layer of protection for authentication.

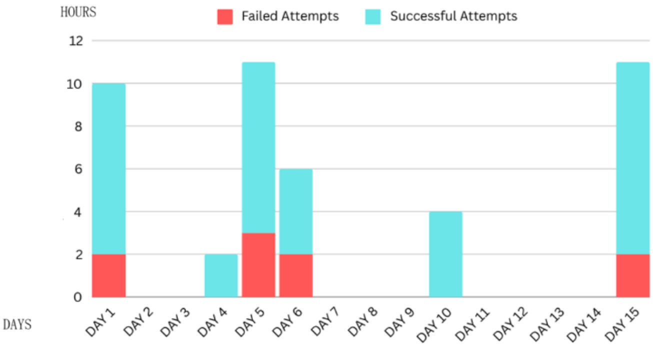


Figure 9: Successful device connection chart to twingate

The graph tracked the login success rate for EC2 resources after users authenticated through the Twingate client as shown in Figure. 9 above. The insurance of authorized users with verified devices can only access the resources, showing the effectiveness of Zero Trust because of its high success rate. If a user's device and environment are verified and authorized by the admin, they would always gain access—connection and only failed when there were configuration issues, which is rare if the user is properly trained in MFA. This supports the thesis by showing how Zero Trust efficiently manages access control to EC2 resources.

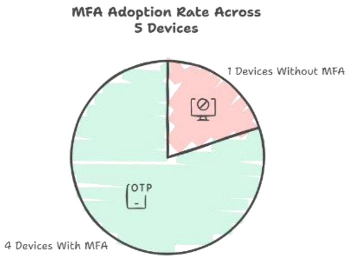


Figure 10: MFA Adoption Rates

The graph tracks the MFA adoption rate among devices logged into the Twingate client as shown in Figure. 10 above, showing that 4 out of 5 devices successfully used MFA to access AWS resources in the home network. Robust authentication is relied on the MFA usage rate and in this research every user needs to go through MFA because it is crucial and required also it is an extra layer of security in Zero Trust. This directly supported the thesis by addressing authentication issues, highlighting how Zero Trust effectively secures the home network by verifying both users and devices through MFA.

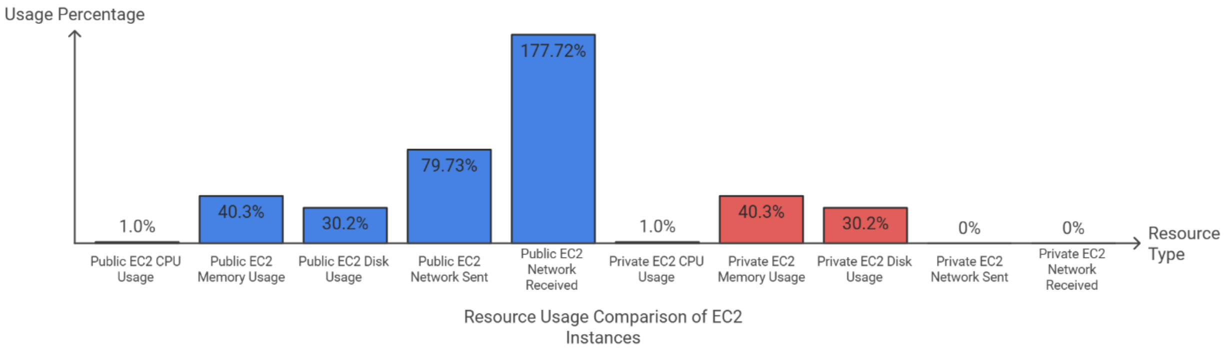


Figure 11: Public and Private usage of resources with EC2

Twingate worked best with a public EC2 instance as shown in above Figure. 11. It acts as a management console for Zero Trust and provides a segmented network mechanism that overcomes the need of using VPN. It is best suited because even if the resource is on public IP for instance EC2 it prevents it from exposing its IP in public by providing remote access. On the other side private EC2 instance comes with same security protocols but the limitation here is access to Internet that is why Public EC2 contain more value in real world as compared to Private EC2.

6.4. Experiment 04 From Confidentiality

Twingate by default provided HD Encryption to the connected devices but there was a need to integrate an extra layer of encryption to storing the data in the EC2 that was integrated with KMS this research automated it too in this research but to compare the KMS and default Encryption time and other information this research did it manually with our terminal where this research generated and uploaded a large text file to our EC2 and measure the time of Encryption and Decryption as shown in Figure. 12. This research have also analyzed ACL for some of our uploaded file on S3 bucket that does not allowed any other user to access those files through the internet for example User uploaded an html file so it only be visible to the specific user with privileges to view it rather than any other person in the same network even if the file itself was visible to that user, but not accessible to view as shown in Figure 13.

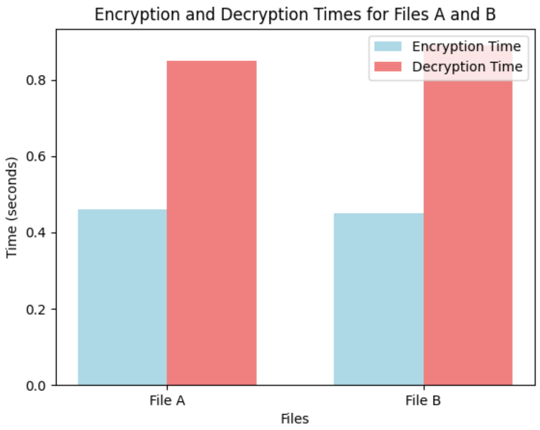
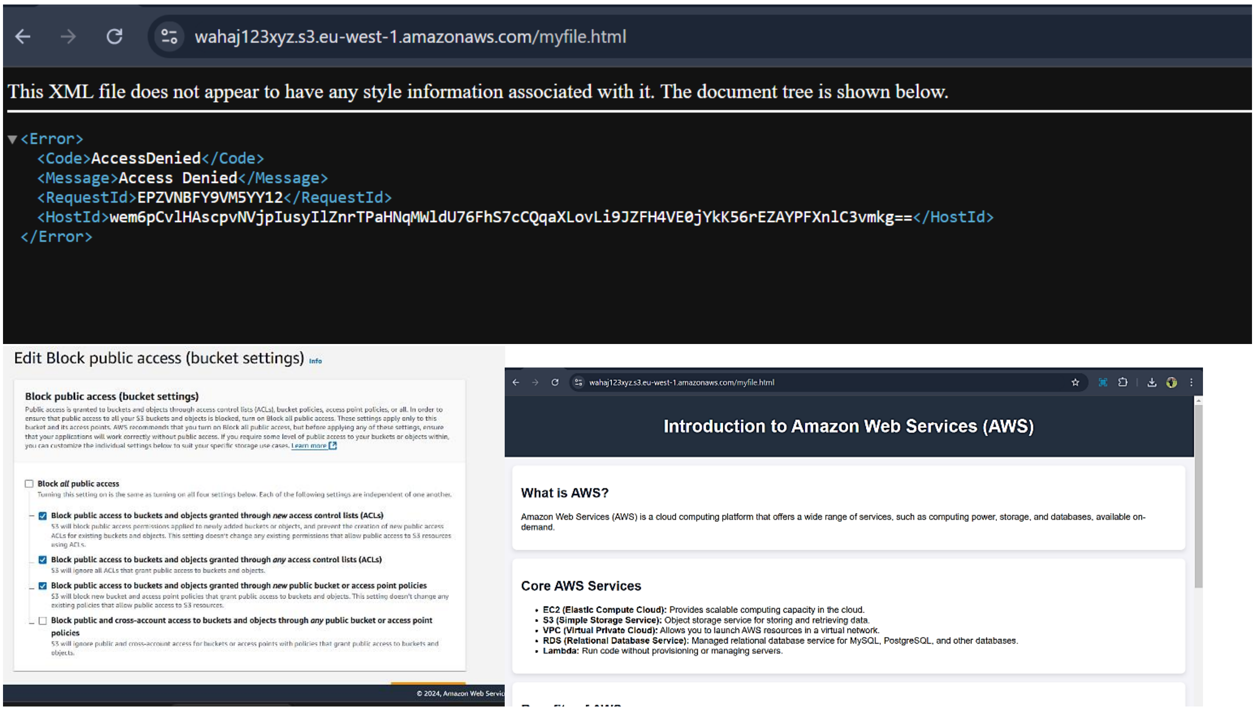


Figure 12: Time taken to encrypt and decrypt

As shown in Figure. 12 File A which is a large text file with a size of 30MB has (KMS encryption) and File B with same type and size as A (normal upload) clearly shows that both have similar encryption and decryption times. Despite the added security of KMS, the performance difference is minimal, made KMS a viable option for secure data handling in a Zero Trust environment without significant impact on performance. TXT file has chosen is to evaluate the encryption and decryption level with a large text size to get results in the minimum possible time.



wahaj123xyz.s3.eu-west-1.amazonaws.com/myfile.html

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>AccessDenied</Code>
   <Message>Access Denied</Message>
   <RequestId>EPZVNBFY9VM5YY12</RequestId>
   <HostId>wem6pCvlHAscpvNVjpIusyIlZnrTPaHNqMWldU76FhS7cCQqaXLovLi9JZFH4VE0jYkK56rEZAYPFXnlC3vmkg==</HostId>
 </Error>
```

Edit Block public access (bucket settings) Info

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

□ **Block *all* public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- ☑ **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
  S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- ☑ **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
  S3 will ignore all ACLs that grant public access to buckets and objects.
- ☑ **Block public access to buckets and objects granted through *new* public bucket or access point policies**
  S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- □ **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
  S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

© 2024, Amazon Web Servic

wahaj123xyz.s3.eu-west-1.amazonaws.com/myfile.html

Introduction to Amazon Web Services (AWS)

What is AWS?

Amazon Web Services (AWS) is a cloud computing platform that offers a wide range of services, such as computing power, storage, and databases, available on-demand.

Core AWS Services

- **EC2 (Elastic Compute Cloud):** Provides scalable computing capacity in the cloud.
- **S3 (Simple Storage Service):** Object storage service for storing and retrieving data.
- **VPC (Virtual Private Cloud):** Allows you to launch AWS resources in a virtual network.
- **RDS (Relational Database Service):** Managed relational database service for MySQL, PostgreSQL, and other databases.
- **Lambda:** Run code without provisioning or managing servers.

Figure 13: Access Control list with least privileges control for user files

6.5. Discussion

This research focused on addressing the issues in confidentiality and authentication with AWS services in a home network model by validating the integration of automated Zero Trust framework. Each experiment was designed to validate and test the proposed approach and to evaluate the effectiveness of the security in a home network. Key findings validate that deploying through automation Infrastructure as Code reduced the chances of human error up to 90% as shown in Fig. 9 because before deployment the user validate the code and fix the issues before deploying it and deployment time has reduced as shown in Figure. 6 because manual configuration takes hours while IaC is doing the same thing in minutes. S3 bucket data achieved a 100% encryption using AES256 by default and we further evaluate its enhancement using KMS that takes a few second to encrypt and decrypt the file as shown in Figure 12. On the authentication side MFA and biometric mechanism reduced the chances of unauthorized access since it is not only relying on the user but its device and IOS environment verification too as shown in Figure. 8.

6.5.1. Strengths and Outcomes:

- By comprehensive automation, reduction in downtime of deployment, chances of Human intervention and errors have reduced.

- Confidentiality challenges evaluated effectively by automated Encryption mechanism in Zero Trust and further enhanced to KMS. S3 bucket data achieved a 100% encryption using AES256 by default and we further evaluate its enhancement using KMS that takes a few second to encrypt and decrypt.
- Access control list restrict user files with least privilege control and prevent to get exposed even in public domain or public EC2 and logs for user activities are automated to get stored in S3 log bucket.
- On the authentication side MFA and biometric mechanism reduced the chances of unauthorized access since it is not only relying on the user but its device and IOS environment verification too before giving access to the user

6.5.2. **Limitations**:

- Automation and Artificial Intelligence could enhance policy enforcement and decision making but also lead to advanced resources requirement, complexity in code, and bugs in setup.
- For a home user with lack of knowledge in coding it might be challenging to study and update policy settings.
- A single home network environment made testing limited wide and large scale of testing could expose more insights.

6.5.3. **Suggestions for Improvement**:

- Hybrid encryption techniques with authentication should be used to validate the enhancement of confidentiality and authentication to provide better defense and resilience against threats.
- For policy management there should be simplified user interface with proper documentation that helps users to configure and understand better in their Infrastructure.
- Develop in house solution instead of using third party tool for the home network.

7. Conclusion And Future Work

To conclude this research, evaluate the issues being resolved related to authentication and confidentiality in a home network environment through the automation of Zero Trust with the services of AWS. The principle of “never trust always verify” integrated using Infrastructure as code with zero trust management console to evaluate the enhancement of authentication because it ensured only verified devices and authenticated user could access the network with its resources after passing the layer of MFA and Biometric. To evaluate the effectiveness of the network AWS resources like EC2 and S3 being used and integrated some access management policies like ACL with encryption techniques like ACL to evaluate the enhancement of confidentiality of data. Twingate serves as the orchestrator of Zero Trust, enabling secure, password less access for pre-approved devices and users through its connectors, while network admins manage access and authentication layers, including MFA and policy enforcement. This whole process not only prevents unauthorized access, reduces the chances of human error, provides segmentation of network but also prevents data leakage.

The implementation and evaluation of a network with Automated Zero Trust Model for the future researcher could be further enhanced to improve confidentiality and authentication challenges by using Technologies related to Quantum and Artificial Intelligence or Machine Learning models for instance Reinforcement Learning and Random Forest are suited for detection mechanism, prediction for anomaly and more polished policies related to access control. To enhance the encryption mechanism Quantum Key Distribution for key exchange securely or Post Quantum cryptography could be used to optimize the confidentiality of the data that contains sensitive information. Moreover, future researchers could test the model in a more complex environment to find its vulnerabilities and fix them to enhance the scalability of the setup. For better confidentiality homomorphic encryption should be implemented. SSH key should be generated by the time of MFA process to provide ease for user to use the resources. Innovation like these could provide much more resilient home network that not only would ensure the advance confidentiality but straight forward confidentiality with extra layer of threats detections and preventions for the users in the network.

8. References

Ahmadi, S. (2024) *'Zero trust architecture in cloud networks: application, challenges and future opportunities'*. Rochester, NY. Available at: https://papers.ssrn.com/abstract=4725283 (Accessed: 6 December 2024).

AWS Key Management Service - AWS Key Management Service (n.d.). Available at: https://docs.aws.amazon.com/kms/latest/developerguide/overview.html (Accessed: 5 December 2024).

AWS resources and shared VPC subnets - Amazon Virtual Private Cloud (n.d.). Available at: https://docs.aws.amazon.com/vpc/latest/userguide/vpc-sharing-service-behavior.html (Accessed: 10 December 2024).

Buck, C. *et al.* (2021) 'Never trust, always verify: A multivocal literature review on current knowledge and research gaps of zero-trust', *Computers & Security*, 110, p. 102436. doi:10.1016/j.cose.2021.102436.

Build security into your network's DNA: The zero trust... (n.d.) Forrester. Available at: https://www.forrester.com/report/build-security-into-your-networks-dna-the-zero-trust-network-architecture/RES57047 (Accessed: 5 December 2024).

Could greater reliance on cloud computing mean a cybersecurity risk... (2023) Nomios UK. Available at: https://www.nomios.co.uk/news-blog/could-greater-reliance-on-cloud-computing-mean-a-cybersecurity-risk-for-businesses/ (Accessed: 5 December 2024).

Device security controls | Docs (n.d.). Available at: https://www.twingate.com/docs/device-controls-use-case (Accessed: 5 December 2024).

Eidle, D. *et al.* (2017) 'Autonomic security for zero trust networks', in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pp. 288–293. doi:10.1109/UEMCON.2017.8249053.

Gwilliam, A. (2022) 'ZTNA vs. VPN: What's the difference?', *JumpCloud*, 25 May. Available at: https://jumpcloud.com/blog/ztna-vs-vpn (Accessed: 10 December 2024).

How to use Terraform with AWS and Twingate | Docs (n.d.). Available at: https://www.twingate.com/docs/terraform-aws (Accessed: 6 December 2024).

Implementing a Zero Trust Architecture (2023). *NIST Special Publication (SP) 1800-35 (Withdrawn)*. National Institute of Standards and Technology. Available at: https://csrc.nist.gov/pubs/sp/1800/35/2prd (Accessed: 5 December 2024).

Modi, C. *et al.* (2013) 'A survey of intrusion detection techniques in cloud', *Journal of Network and Computer Applications*, 36(1), pp. 42–57. doi:10.1016/j.jnca.2012.05.003.

Omier, E. (2022) 'Why the castle and moat approach to security is obsolete', *The New Stack*, 21 June. Available at: https://thenewstack.io/why-the-castle-and-moat-approach-to-security-is-obsolete/ (Accessed: 5 December 2024).

Sarkar, S. *et al.* (2022) 'Security of zero trust networks in cloud computing: a comparative review', *Sustainability*, 14(18), p. 11213. doi:10.3390/su141811213.

Siddhartha, C.S. *et al.* (2023) 'Enhancing home security: user authentication techniques for home automation', in *2023 International Conference on Inventive Computation Technologies (ICICT)*, pp. 1166–1171. doi:10.1109/ICICT57646.2023.10133997.

The future of VPN: Transitioning from traditional VPNs to Zero Trust (n.d.). Available at: https://www.archonsecure.com/blog/the-future-of-vpn-transitioning-from-traditional-vpns-to-zero-trust (Accessed: 10 December 2024).

Tuyishime, E. *et al.* (2024) 'Online laboratory access control with zero trust approach: Twingate use case', in *2024 16th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*. Iasi, Romania: IEEE, pp. 1–7. doi:10.1109/ECAI61503.2024.10607562.

Two-factor authentication | Docs (n.d.). Available at: https://www.twingate.com/docs/two-factor-authentication-security-policies (Accessed: 5 December 2024).

VPN replacement | Docs (n.d.). Available at: https://www.twingate.com/docs/vpn-replacement-use-case (Accessed: 5 December 2024).

Vundavalli, C. (2024) 'Implementing Zero Trust Architecture with Terraform in multi-cloud environments', *Medium*, 13 February. Available at: https://medium.com/@chaitanyavundavalli/implementing-zero-trust-architecture-with-terraform-in-multi-cloud-environments-0895bfc23433 (Accessed: 5 December 2024).

Wang, W. *et al.* (2024) 'Applying transparent shaping for zero trust architecture implementation in AWS: a case study'. *arXiv*. doi:10.48550/arXiv.2405.01412.

Weinberg, A.I. and Cohen, K. (2024) 'Zero Trust implementation in the emerging technologies era: a survey', *Complex Engineering Systems*, 4(3). doi:10.20517/ces.2024.41.

What is Zero Trust? A model for more effective security (n.d.) CSO Online. Available at: https://www.csoonline.com/article/564201/what-is-zero-trust-a-model-for-more-effective-security.html (Accessed: 5 December 2024).

ZeroKMS: Zero Trust Key Management (n.d.). Available at: https://cipherstash.com/products/zerokms (Accessed: 10 December 2024).

ZTA (n.d.) castle-and-moat. Cloudflare. Available at: https://www.cloudflare.com/en-gb/learning/access-management/castle-and-moat-network-security/.