NOTRE DAME UNIVERSITY
B A N G L A D E S H
The Competence to See and the Courage to Act

Project Report

Course Title: Computer Networks Lab

Course Code: CSE-3104

Project Title: A comprehensive design for an IoT-based private network with public access.

Submitted By:

Member's Name	ID No	Batch
Sajid Hossain Khan	0692210005101007	CSE-18
Md. Shohel Rana	0692210005101016	CSE-18

Submitted To:

Dr. Fernaz Narin Nur

Professor,

Department of CSE, MIST

Date of Submission: 27/11/2024

Table of Contents

1. Introduction

In modern network environments, IoT devices are prevalent, often operating within private networks to enhance security and minimize external threats. This project demonstrates a unique solution to provide controlled public access to a private IoT network while maintaining data flow integrity and security through port forwarding and tunneling.

Our design enables external users to manage IoT devices indirectly via a centralized private server. The server acts as an intermediary, ensuring seamless communication between public clients and private IoT devices.

2. Objectives

- Establish a private network with dynamic private IP addresses for IoT devices and PCs.
- Ensure the IoT devices continuously send updates to a private server.
- Allow external access to the server for IoT device control through port forwarding.
- Use tunneling to virtually connect remote areas (Area 10 and Area 20) for secure communication.
- Demonstrate the use of loopback IPs for testing and routing flexibility.

3. Network Design Overview

3.1 Network Configuration:

- **NAT Router:**
  - **Dynamic IP Allocation:** Six dynamic public IPs are allocated for external communication.
  - **Private-to-Public Translation:** NAT ensures devices within the private network can communicate externally without direct access.
- **Private Network:**
  - IoT devices and PCs operate with private IP addresses.
  - Devices in the private network cannot be directly pinged from the outside world.
- **Server:**
  - Hosts data collected from IoT devices.
  - Acts as a central node for external access via port forwarding.
- **External Access:**
  - Public clients access the server using port-forwarded rules to control IoT devices indirectly.

3.2 Tunneling:

To connect Area 10 and Area 20 virtually:

- **Virtual Tunnel:** A secure tunneling protocol (e.g., GRE, IPsec) was configured to bridge networks that are physically separate.
- **Loopback IPs:** Used as endpoints for the tunneling configuration, enabling seamless routing without dependency on physical interfaces.

4. Implementation Details

4.1 Port Forwarding Configuration

- **Purpose**: To allow public access to specific server functionalities while maintaining network security.
- **Configuration**:
  - The server is set up to forward essential ports:
    - **Port 80** for HTTP traffic.
    - **Port 22** for secure SSH access.
  - **NAT Router Rules**:
    - Rules were added to the router, ensuring that requests from external users are forwarded to the appropriate private server.
- **Result**: External clients can access the server's services without compromising the private network.

4.2 IoT Device Communication

- **Purpose**: To enable real-time communication between IoT devices and the server for continuous data exchange.
- **Protocol Used**:
  - IoT devices utilize the **MQTT** protocol, ideal for lightweight, efficient data transfer.
- **Process**:
  - IoT devices send periodic updates to the server, ensuring continuous monitoring and control. o The server processes these updates and makes them accessible to public clients via port-forwarded connections.

4.3 Tunneling Setup

- **Virtual Tunnel Creation**:
  - A **GRE (Generic Routing Encapsulation)** tunnel was set up to connect **Area 10** and **Area 20**, facilitating seamless data transmission between the areas.

- **Routing Configuration**:
    - Tunnel traffic is routed via **loopback IP addresses**, ensuring redundancy and efficient traffic management.
- **Security**:
    - **IPsec Encryption** was applied to the tunnel to secure communication, ensuring **data confidentiality** and **integrity** during transmission.

4. Technical Implementation

This section outlines the technical aspects of the project, including NAT configurations, tunneling, OSPF routing, and port forwarding.

1. Network Address Translation (NAT)

1.1 Dynamic NAT Configuration

Dynamic NAT is used to translate private IPs into a pool of public IPs, enabling multiple devices to communicate with the external world.

Defining Public and Private Pools:

```
ip nat pool PublicPool 203.0.113.1 203.0.113.6 netmask 255.255.255.248
access-list 1 permit 192.168.10.0 0.0.0.255
ip nat inside source list 1 pool PublicPool
```

Interface Configuration:

```
interface FastEthernet0/0
 ip address 192.168.10.1 255.255.255.0
 ip nat inside
!
interface FastEthernet0/1
 ip address 203.0.113.10 255.255.255.0
 ip nat outside
```

1.2 Static NAT Configuration

Static NAT is configured to allow specific private servers to be accessible externally.

Example Mapping:

- Private IP: `192.168.10.2` → Public IP: `198.51.100.20`

CLI Commands:

```
ip nat inside source static 192.168.10.2 198.51.100.20
```

2. Tunneling

Tunneling connects two separate network areas (Area 10 and Area 20) using a GRE (Generic Routing Encapsulation) tunnel. This enables secure and private communication.

- **Loopback IPs:**
    - **Area 10:** 10.10.10.1
    - **Area 20:** 20.20.20.1

Router Configuration for Tunnel:

```
interface Tunnel0
 ip address 192.168.1.1 255.255.255.0
 tunnel source <source-ip>
 tunnel destination <destination-ip>
 tunnel mode gre ip
!
interface Loopback0
 ip address 10.1.1.1 255.255.255.255
!
```

3. OSPF (Open Shortest Path First)

OSPF is used as the dynamic routing protocol for this project to ensure efficient data exchange between areas.

- **OSPF Areas:**
    - **Area 10:** Internal devices.
    - **Area 20:** IoT devices.

Router Configuration:

```
router ospf 1
 network 192.168.10.0 0.0.0.255 area 10
 network 192.168.20.0 0.0.0.255 area 20
!
interface FastEthernet0/0
 ip address 192.168.10.1 255.255.255.0
!
interface FastEthernet0/1
 ip address 192.168.20.1 255.255.255.0
```

4. Port Forwarding

Port forwarding is used to control IoT devices by accessing the private server from the public network.

Port Mapping Example:

- **IoT Server Private IP:** 192.168.10.5
- **External Port:** 8080

Router NAT Configuration:

```
ip nat inside source static tcp 192.168.10.5 80 203.0.113.5 8080
interface FastEthernet0/0
 ip address 192.168.10.1 255.255.255.0
 ip nat inside
!
interface FastEthernet0/1
 ip address 203.0.113.1 255.255.255.0
 ip nat outside
```

Dynamic NAT Public and Private Pools

Pool Type	Start IP	End IP	Subnet Mask
Public Pool	203.0.113.1	203.0.113.6	255.255.255.248
Private Pool	192.168.10.0	192.168.10.255	255.255.255.0

5. Pictorial Overview

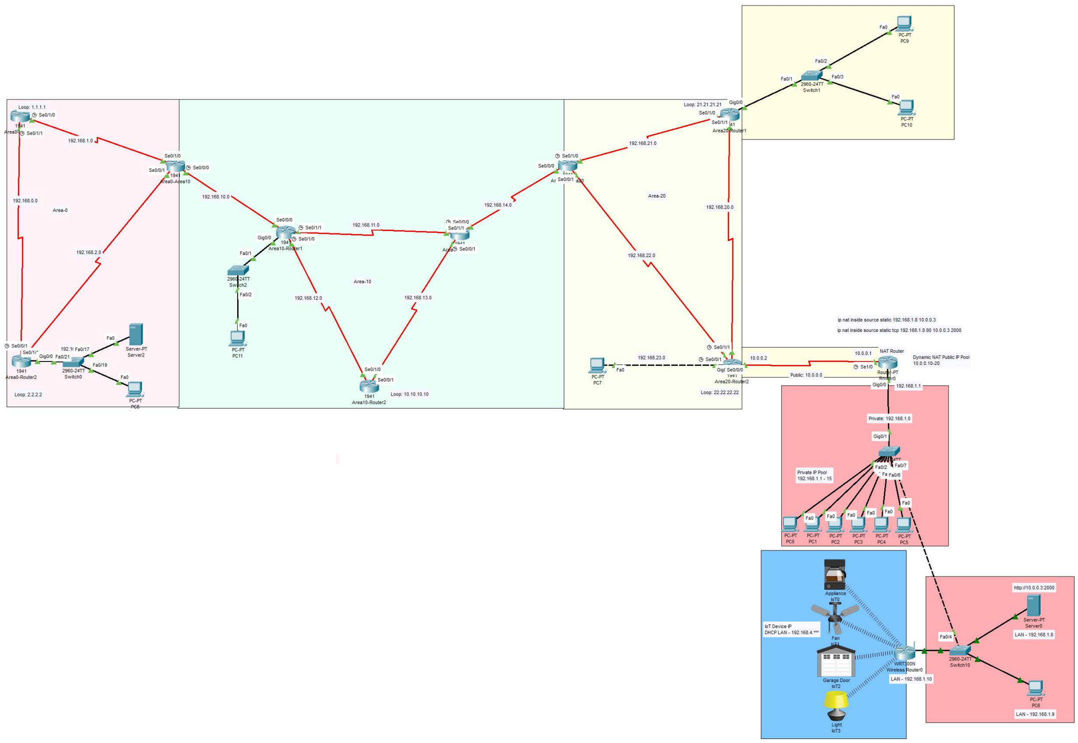

Figure 1: Network Setup

6. Results

- **Private Network Security:** Devices within the private network remained inaccessible directly from the outside world, ensuring security.
- **External Access to IoT Devices:** Public users could control IoT devices through the private server via port forwarding.
- **Continuous IoT Updates:** IoT devices successfully sent real-time updates to the server without requiring external pings.
- **Tunneling Functionality:** Data exchange between Area 10 and Area 20 occurred seamlessly, validating the virtual tunnel's effectiveness.

7. Challenges and Solutions

Challenge	Solution
NAT blocking direct external access to the server	Configured port forwarding to enable controlled access.
Ensuring secure communication between Area 10 and Area 20	Used IPsec for encrypting tunneling traffic.

8. Features

- **Controlled Public Access:** Enabled public users to manage IoT devices indirectly, maintaining security.
- **Seamless IoT Communication:** Continuous data flow ensured real-time updates without direct external pings.
- **Virtual Tunneling:** Connected remote areas virtually, demonstrating a scalable networking solution.

9. Conclusion

This project highlights an innovative approach to designing a secure and efficient IoT-based private network with public access. By combining NAT, port forwarding, and tunneling technologies, we created a scalable and secure system that can be implemented in real-world scenarios, such as smart homes or industrial IoT networks.