Premier University

Chattogram

Project Proposal

Design and Simulation of an IPv6 Smart City IoT Network

with Quality of Service and Resilient Routing

Submitted by

Name	ID
MD Nishadul Islam Chy Shezan	0222220005101014
MD Sakib	0222220005101019
Rimjhim Dey	0222220005101039

Section: A

Batch: 42

Session: Spring 2025

Supervisor

Dr. Shahid Md. Asif Iqbal
Professor
Department of Computer Science and Engineering
Associate Dean
Faculty of Engineering

July 2025

Project Overview

In today's cities, a huge number of sensors collect all kinds of information about traffic, air quality, and even how full the bins are. Most networks in use still run on IPv4, which isn't great at handling so many devices or making sure urgent messages get delivered first. With this project, we're building a smart city network model in Cisco Packet Tracer that uses IPv6, sets up Quality of Service (QoS) for important data, and tests out edge (fog) routers for faster local processing. We want to show how these networking upgrades can make city sensor systems work better, especially during failures.

Problem Statement

Sensors are everywhere in modern cities, gathering information on everything from air quality to traffic flow to even the amount of trash in the bins. The smooth operation of a city depends on this knowledge. Many of the networks in use today, however, are still based on the antiquated IPv4 protocol, which is unable to meet the increasing demands for scalability and urgent communications, such as incident alerts. This project aims to develop a smart city network prototype in Cisco Packet Tracer that utilizes IPv6, integrates Quality of Service (QoS) mechanisms to prioritize essential traffic, and implements edge (fog) routers to speed up processing at the network's edge. Through this initiative, we want to show how these modern networking techniques can help cities get the most out of their sensor data, especially during failures or emergencies.

Objectives

3.1 Design a smart city network in Cisco Packet Tracer with layers for access, distribution, and core.

3.2 Implement IPv6 everywhere so there's no risk of address exhaustion.

3.3 Set up Quality of Service to prioritize traffic, especially emergencies.

3.4 Use edge (fog) routers to process sensor data closer to where it's collected.

3.5 Test network failover using HSRP and EIGRP for IPv6.

3.6 Keep IoT, public Wi-Fi, and admin devices on separate VLANs, protected by ACLs.

These objectives guide our project, making sure we focus on both the technical setup and real-life needs of a modern smart city network.

Key Benefits and Limitations

Strengths

4.1 With IPv6, we can easily connect a lot more devices without the hassle of running out of addresses.

4.2 Implementing QoS makes sure that urgent messages, like emergency alerts, won't get tangled up in network traffic.

4.3 By utilizing VLANs and ACLs, we can keep different areas of the network isolated and more secure.

4.4 Setting up edge routers near the sensors allows for quicker data processing and helps prevent overloading the main network.

Challenges

4.1 Packet Tracer is a simulator, so it can't capture every detail you'd see in a real network.

4.2 Wireless issues and interference can't really be tested in this environment.

4.3 The MQTT protocol in Packet Tracer is basic compared to what's used in practice.

4.4 NAT64 in the simulation doesn't always work like it does on actual devices.

This section shows that while our project has some strong points, we're also aware of the real-world and simulation limits we have to work within.

Features of a Complex Problem (5 out of 7 Criteria Met)

Our project hits several points that make it a challenging, real-world networking problem:

5.1 **Conflicting Requirements:** We need the network to deliver important data quickly, but also keep it efficient for regular traffic.

5.2 **Multiple Stakeholders:** The network is used by emergency teams, city staff, and the public.

5.3 **Depth of Analysis:** Routing, IPv6 setup, QoS, and failure testing all take planning and technical work.

5.4 **Extensive Knowledge Base:** We apply what we learned about IoT, protocols, security, and monitoring.

5.5 **Interdependence:** Sensors, edge routers, and core routers all have to work together for the network to function.

This list highlights how the project meets several criteria for complex engineering tasks. It's not just about building the network, but also thinking about how all the parts fit and work together.

Methodology

6.1 Build the network in Cisco Packet Tracer, including all device layers.

6.2 Set up IPv6 addressing and subnets.

6.3 Configure QoS so emergency data is prioritized.

6.4 Use EIGRP for IPv6 and HSRP for redundancy.

6.5 Separate devices with VLANs and secure with ACLs.

6.6 Test the setup by simulating failures and heavy traffic.

These steps give a clear path for us to follow, from planning out the network to actually testing if it works under different conditions.

Expected Outcome

7.1 A working smart city network that supports many devices.
7.2 Data like emergency alerts is delivered quickly, even if something fails.
7.3 The network can recover from problems automatically.
7.4 Each part of the network (IoT, public Wi-Fi, admin) is kept separate and secure.

If we achieve these results, we'll have shown that our design can handle smart city demands and is ready for further real-world testing.

Risks and Mitigation

8.1 **Simulation limits:** Packet Tracer can't do everything a real network can. *Mitigation:* Use scripting and MQTT basics to make traffic more realistic.
8.2 **QoS visibility:** Measuring real packet delays is tough in simulation. *Mitigation:* Rely on simulation analytics and verify configuration.
8.3 **IPv6 config errors:** Manual address setup can be error-prone. *Mitigation:* Plan addresses carefully and check twice.

Being aware of these risks helps us prepare solutions in advance, so our project stays on track even if we run into problems.

References

9.1 Cisco Systems. "Packet Tracer Labs and IoT Modules."
9.2 RFC 8200: Internet Protocol, Version 6 (IPv6) Specification
9.3 IEEE 802.1Q VLAN Tagging Standard
9.4 IoT-A Architecture Reference Model, European Commission