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

Submitted to

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

August 2025

Contents

Contents

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. The implementation will showcase IPv6's virtually unlimited addresses, edge computing for reduced latency, and how emergency services maintain priority even under congestion through proper QoS configuration.

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. Current IPv4 limitations include address exhaustion requiring complex NAT, lack of native IoT support, and centralized processing bottlenecks that our hierarchical IPv6 architecture with distributed fog computing will address.

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. The hierarchical design ensures scalability while IPv6 future-proofs the infrastructure for decades of growth. QoS configuration will demonstrate dynamic resource allocation ensuring ambulance dispatch isn't delayed by social media traffic, while edge computing shows measurable response time improvements for local decision-making.

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.

IPv6 provides 340 undecillion addresses, eliminating NAT complexity and improving both performance and troubleshooting. QoS uses DSCP markings to classify traffic into priority levels, guaranteeing bandwidth for life-critical services during peak usage. The security architecture implements defense-in-depth with VLAN Layer 2 isolation and ACL Layer 3 policies, creating multiple security boundaries.

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 Some services like QoS and email alerts are basic compared to real-world implementations.

Despite limitations, Packet Tracer effectively demonstrates core concepts and validates architectural decisions without vendor-specific complexity. While we can't simulate real wireless interference or advanced QoS mechanisms, we can still show traffic prioritization impact on service delivery. The simplified implementations focus attention on fundamental principles that scale to production environments.

Features of a Complex Problem

This project meets all the required aspects of a complex engineering problem:

Criteria	How Our Project Meets It
Conflicting Requirements	We must deliver emergency alerts quickly without disrupting normal traffic.
Multiple Stakeholders	IoT sensors, city staff, admin PCs, and public users all share the same network.
Depth of Analysis	Involves IPv6 setup, VLANs, ACLs, failover, and basic QoS.
Extensive Knowledge Base	Uses skills from IPv6 addressing, routing, ACLs, VLANs, and services like SMTP and HTTP.
Interdependence	Sensors, servers, core routers, and admin devices depend on each other to function correctly.
Nobility (Public Impact)	Supports cleaner, safer city operations by helping staff respond faster to issues.
Innovation	Combines IPv6 and basic fault-tolerant design in a small smart city IoT network.

The complexity extends to real-world trade-offs like balancing public service utilization against emergency bandwidth guarantees through sophisticated traffic management. Component interdependence means a malfunctioning sensor could trigger false alerts, overwhelming systems and masking genuine emergencies. Our solution addresses these through careful design, redundancy, and comprehensive testing protocols.

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.

Following industry best practices, we'll implement hierarchical IPv6 allocation using /48 for sites and /64 for subnets with EUI-64 for automatic addressing. QoS will use a four-tier model: network control, emergency services, standard IoT, and best-effort public traffic. EIGRP metrics will be optimized for IoT patterns while HSRP provides sub-second gateway failover, with each phase validated through incremental testing.

Device List

To keep the design realistic and manageable in Packet Tracer, we will use a minimal set of devices:

7.1 2 core routers for backbone connectivity and redundancy (HSRP)
7.2 2 distribution switches to connect core and access layers
7.3 2 access switches for end devices
7.4 4–6 IoT end devices (e.g., smart lights, pollution sensors, smart bins)
7.5 2 admin PCs (one for monitoring, one for email alerts)
7.6 1 email server (SMTP) for sending alerts
7.7 2 wireless access points for public and private Wi-Fi

The device selection balances concept demonstration with manageable complexity, using high-performance routers for advanced features and distribution switches for inter-VLAN routing. IoT devices simulate various patterns: periodic updates from lights, continuous streaming from sensors, and event-triggered bin alerts. Dual admin PCs demonstrate monitoring and alert roles while wireless APs showcase segmentation between public and administrative access.

Feature List (What We Will Demonstrate)

The following features will be configured and tested in the network:

8.1 IPv6 addressing and routing across all devices
8.2 VLAN segmentation for IoT devices, admin PCs, and public Wi-Fi
8.3 HSRP failover on core routers for high availability
8.4 ACLs to control access between VLANs
8.5 Basic QoS: give priority to emergency messages over normal traffic
8.6 Email alerts sent through a simple SMTP server in Packet Tracer
8.7 Ping and basic service tests to confirm connectivity and failover

Each feature represents critical smart city components, with IPv6 using both SLAAC and DHCPv6 for flexible address management. VLAN segmentation implements 802.1Q tagging for separate broadcast domains improving security and performance. HSRP uses authentication and interface tracking for intelligent failover while ACLs demonstrate granular traffic control, with metrics showing latency improvements under various scenarios.

Communication Plan

We will keep communication simple, using the built-in services available in Packet Tracer:

9.1 **IPv6 traffic:** All devices will communicate using IPv6-only addressing.

9.2 **Email (SMTP):** Admin PCs will receive email alerts from the central email server when critical events occur.

9.3 **Web interface (HTTP):** Admin PCs will use a basic HTTP service for status pages hosted on the server.

9.4 **ICMP pings:** Used to test connectivity and failover when simulating failures.

Example flow: A smart bin reaches full capacity → triggers a message → email server sends a simple SMTP alert to admin → admin PC can log in via HTTP to check system status.

The architecture uses a scalable publish-subscribe pattern where IoT devices publish events and administrators receive notifications, easily expanding as sensors are added. IPv6-only eliminates dual-stack complexity while demonstrating next-generation capabilities. The HTTP dashboard provides device status and event history, simulating real network operations centers.

Test Plan

We will run simple tests to verify each feature:

10.1 Simulate a router failure and confirm HSRP takes over.

10.2 Send traffic during an ”emergency” and check QoS priority.

10.3 Trigger a sensor event and verify SMTP email alert is received.

10.4 Ping all devices from admin PC to confirm IPv6 connectivity and ACL behavior.

Testing encompasses functional validation and performance benchmarking with sub-second failover targets and minimal packet loss during transitions. QoS tests will generate competing flows measuring latency, jitter, and throughput per traffic class, validating emergency traffic consistency under congestion. End-to-end sensor alerts will measure total response time while connectivity tests validate both authorized communications and proper blocking of unauthorized attempts.

Timeline

The proposed timeline for the completion of the project is outlined below:

Weeks	Phase	Key Activities/Outcomes
Weeks 1–2	Initial Literature Review	Meet with supervisor; gather background information; identify key research papers
Weeks 3–4	Project Planning & Project Proposal Submission	Detailed project planning, finalize objectives and deliverables, and submit the official project proposal
Weeks 5–6	Network Design & Topology	Design network layers in Cisco Packet Tracer, select devices, create network diagrams
Weeks 7–8	Configuration: IPv6, VLANs, QoS	Configure IPv6 addressing and subnets, set up VLANs, implement Quality of Service policies
Weeks 9–10	Redundancy, Security & Testing	Set up EIGRP for IPv6, HSRP, and ACLs; test failover, segmentation, and security features
Weeks 11	Documentation & Final Report Preparation	Compile results, write and format the report, finalize project documentation
Week 12	Project Submission	Submit the complete project and present findings as required

The timeline includes buffer periods for challenges and iterative refinement, with each phase building on previous work through overlapping integration periods. Literature review focuses on recent IPv6 deployments and smart city architectures while design produces detailed diagrams and configuration templates. Implementation allows incremental validation preventing cascading issues, with final weeks ensuring comprehensive documentation meeting academic standards.

Expected Outcome

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

The completed project delivers a functional smart city prototype demonstrating enterprise features as a reference architecture with documented best practices. Performance metrics validate handling of typical traffic patterns while maintaining quality during adverse conditions. The implementation provides hands-on experience with critical next-generation technologies, preparing team members for network engineering and IoT system design careers.

Risks and Mitigation

12.1 **Simulation limits:** Packet Tracer can't do everything a real network can. *Mitigation:* Focus on core features and test configs carefully.

12.2 **QoS visibility:** Measuring real packet delays is tough in simulation. *Mitigation:* Use configuration verification and counters.

12.3 **IPv6 config errors:** Manual address setup can be error-prone. *Mitigation:* Plan addresses carefully and check twice.

Additional risks include coordination challenges addressed through weekly syncs and collaborative configuration management tools. Technical knowledge gaps will be resolved via targeted learning and peer mentoring. Version control principles maintain configuration backups before changes, with ongoing risk assessment adapting strategies based on encountered challenges throughout the project lifecycle.

References

13.1 Cisco Systems. "Packet Tracer Labs and IoT Modules."
13.2 RFC 8200: Internet Protocol, Version 6 (IPv6) Specification
13.3 IEEE 802.1Q VLAN Tagging Standard
13.4 Cisco Networking Academy: Packet Tracer Services (SMTP, HTTP)