Appendix B: Montréal Smart City Pilot System (MSCPS) Deployment: Implementation Details

Contents

Abstract

This purpose of this appendix is to outline the physical smart city deployment in Montreal. The appendix describes the various devices deployed, their settings, the overall network setup, and locations. Tests performed include Wi-Fi and ZigBee radio throughput levels and overall network online reliability. Interference sources in downtown Montreal can have significant effects on radio throughput. 802.11ac Wi-Fi radios experienced a drop of 325 Mbits/s to 5 Mbits/s with a 40 MHz channel width and 80 Mbits/s to 70 Mbits/s with 10MHz after deployment. ZigBee radios throughput quickly decreased with distance, and throughput levels with AES encryption are roughly half of their unencrypted speed. The average device was online and visible on the network 95% of the time. Conclusions highlight the importance of carefully selecting devices, designing network architecture, and expecting significantly lower device performance in city deployment than manufacturers advertise.

1 Introduction

To support the vast potential of a smart city, a strong and reliable network and infrastructure is required. With millions of interconnected devices, sensors, and people, every node in the network must be well-managed and secure to prevent accidents.

With constantly evolving communication technologies, a smart city must be capable of synergizing new technologies with old ones. The overall network will be interconnected via ZigBee, Wi-fi, Ethernet, Bluetooth, Fiber Optic, LTE, and more. Given the large number of technologies, cities must be extremely careful in network structure design to avoid data transmission bottlenecks that in some situations could render the entire system too slow to use or too slow to respond to emergencies.

This appendix will detail the findings in a limited-scale IoT smart city deployment. Feasibility and structure review will be discussed to help guide future smart deployments.

1.1 Appendix Organization

The following appendix sections are organized as follows. First, a high-level view of the network architecture is presented including abstract and location setup. Device models, capabilities, and basic explanations are also described. Next, the appendix details the enabled settings on each device during the test-deployment. Subsequently, results of throughput and device reliability will be discussed, followed by conclusions.

2 Network Structure

2.1 Overview

The limited-scale IoT smart city prototype project discussed in this report showcases a structure of interconnected cameras, radars, radios, and other sensors connected through various wired and wireless communication mediums.

In the current scenario, there are more than 50 devices deployed and interconnected including Wi-Fi and ZigBee radios, switches, cameras, RFID readers, and traffic radars. Figure 1 details the interconnectivity of all the devices and their communication methods. Black data speeds represent the max connection speed between devices and orange font is used to designate the average transmitted data rate in current deployment. Naming convention follows the format of a shorthand device-type naming followed by the location code, and an accompanying number. For example, CAM_PKH5_1. Device codes and locations in current deployment can be referenced in Table 1 and Table 2, and a map with locations can be seen in Figure 2. Devices without static locations, such as sensors installed on mobile vehicles will have an omitted location label.

Table 1: Device naming convention codes

Short-hand Code	**Device**
ADPT	Adapter
CAM	Camera
LEV	Level Sensor
LTE	LTE Gateway
RAD	Traffic radar
RFID	RFID reader (includes BLE reader)

TEMP	Temperature sensor
WIFI	Wi-Fi radio
ZIG	ZigBee radio

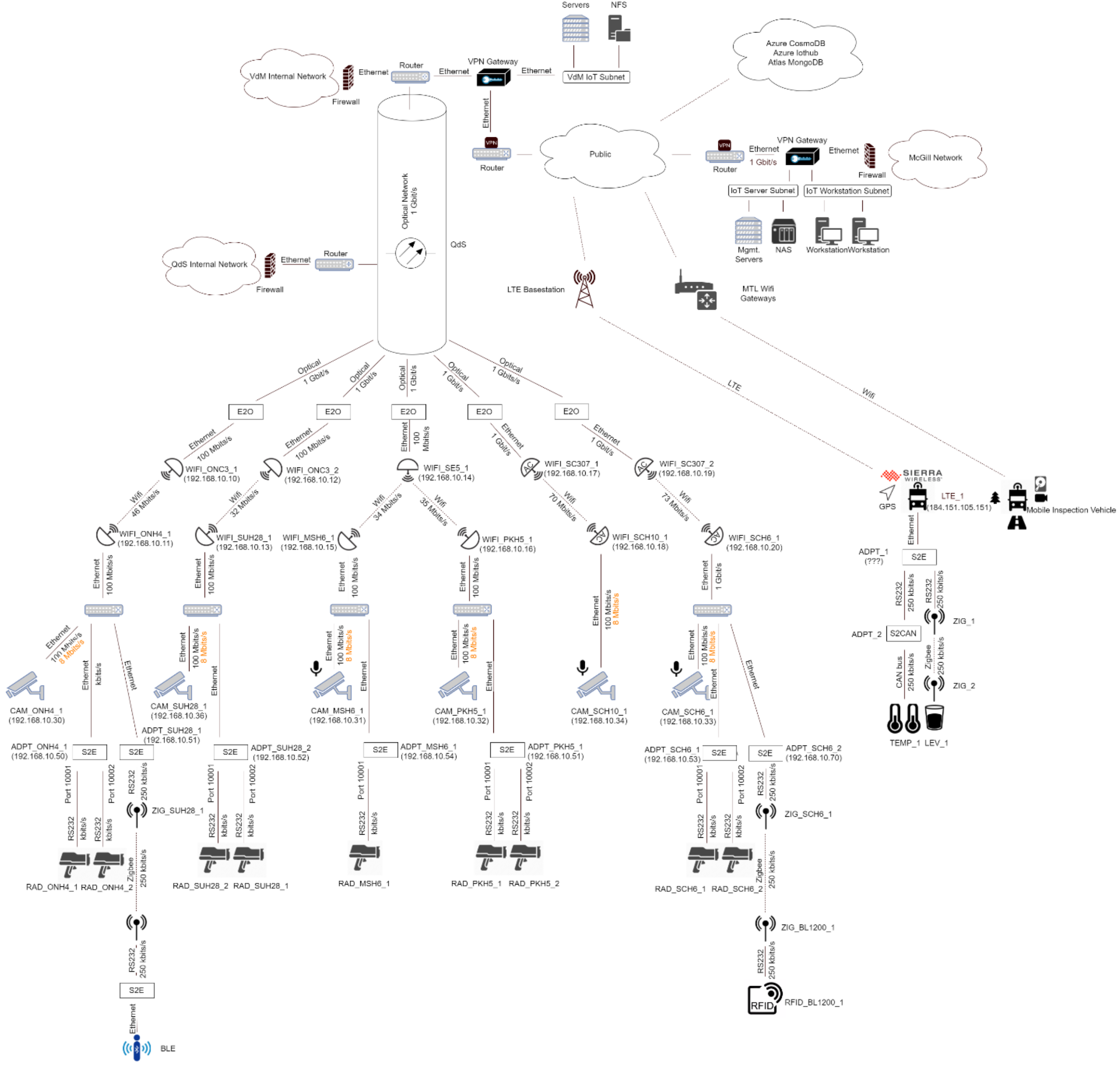


Figure 1: Full architecture diagram of all device

Table 2: Location code meaning

Location Code	Location
BL1200	1200 Rue de Bleury
MSH6	Intersection of Rue Jeanne-Mance and Boulevard de Maisonneuve O
ONC3	32 Rue Ontario O
ONC4	Intersection of Rue Clark and Rue Ontario O

PKH5	Intersection of Rue Jeanne-Mance and Avenue du President-Kennedy
SC307	307 Rue Sainte-Catherine O
SCH10	Intersection of Rue Jeanne-Mance and Rue Sainte-Catherine O
SCH6	Intersection of Rue de Bleury and Rue Sainte-Catherine O
SE5	On Avenue du President-Kennedy
SUH28	Intersection Saint-Urbain and Boulevard de Maisonneuve O

Wi-Fi radios connect various stationary devices across the network to the fiber optic mainframe. Adapters are utilized to bridge communication mediums such as RS232 to Ethernet, which is necessary for connecting devices such as traffic radars. Sensors attached to mobile vehicles either store information on a hard drive and upload when in Montreal Wi-Fi gateway range, or transmit data real-time through LTE.

Devices such as cameras and radars are connected on a private network and via optical fiber cable rather than via LTE connections due to large bandwidth requirements. The bandwidth is too high and the cost would be too expensive for efficient resource use.

To gain access to select devices, including storage devices from the public internet, network address translation is used. Messages and requests are sent to a server with public IP address 132.206.68.25. This management server translates the destination of incoming messages to the device's address on the network and forwards the message further. The same server also provides NTP time synchronization of all devices.

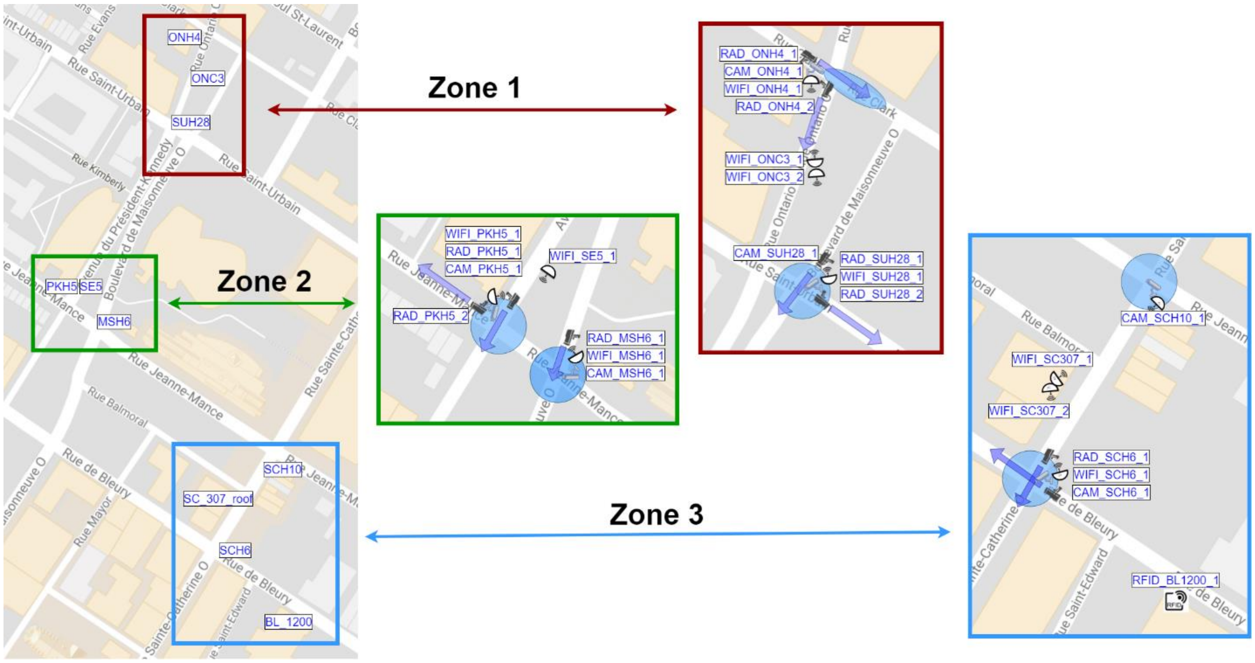


Figure 2: Map of deployment

Device deployment is concentrated in the area of Cartier des Spectacles in three main locations. This is presented in Figure 2, which highlights the sensing devices and radios.

Each zone contains cameras and traffic radars. Zone 3 also contains an RFID reader. Camera viewing zones are shown in blue and traffic radar setup direction is designated by a purple arrow. Not pictured, but present in these deployment zones are Ethernet switches, and Serial-to-Ethernet adapters to attach cameras and radars to the radios.

2.2 Stationary Device Models

This section will detail the device models and parameters used within the prototype deployment. It is split into subsections on communication devices such as Wi-Fi radios and sensors, which include cameras and traffic radars. Communication deployment consists of 11 Ubiquiti Wi-Fi radios, 6 Serial-to-Ethernet adapters from three different vendors, and 6 Digi ZigBee radios. Sensors comprise 6 cameras from three different vendors, 9 Geolux radars, an FEIG RFID reader, and a BLE beacon reader on the stationary network. A picture of a deployed setup can be seen in Figure 3.

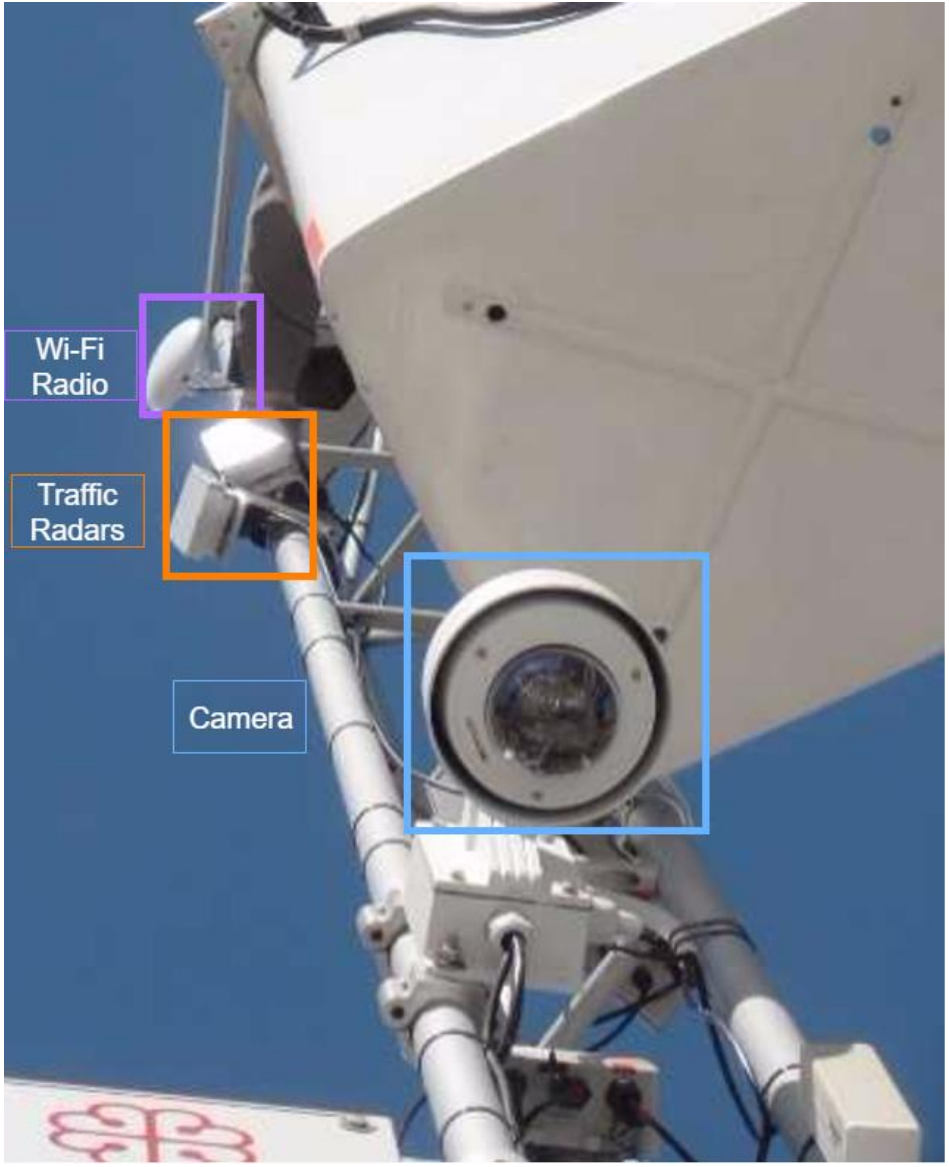


Figure 3: Picture of deployment setup

2.2.1 Communication

Communication devices within the network consist of two main types: wireless and wired. Bridging methods, such as virtual private network tunneling and network address translation, make it possible to link devices into one secure, easy to manage, network. Both communication and bridging methods are discussed in this section.

2.2.1.1 Wireless

Wireless technology is used to deploy devices in areas where it is hard to route Ethernet or Fiber Optic cables. While the wireless connection is often convenient, it is also susceptible to a number of security threats [1].

The danger of wireless communication lies in its visibility. Anyone with a wireless capable device can easily intercept data transmissions if near the propagating source. To prevent information leaks or device cloning attacks, the devices and communication channels must be as safe as possible. This can be achieved by using state of the art encryption [2] discussed further in Appendix G.

Two types of wireless devices are used within the deployment: Wi-Fi and ZigBee.

2.2.1.1.1 Wi-Fi Radios

Wi-Fi radio technology is used to create a high bandwidth wireless communication channel between devices. Wi-Fi functions on the IEEE 802.11ac and IEEE 802.11n protocols. To create a communication Wi-Fi bridge, two radios are required. One acts as an access point (AP) and connects to the main wired network and the other behaves as a station (ST) and is connected to devices. The devices are thereby connected to the main network via the wireless communication channel. Radios can be arranged in point-to-point (PTP) where there is a 1-to-1 radio connection or a point-to-multi-point (PTMP) arrangement where multiple ST send data to one AP.

The Wi-Fi radio deployment in downtown Montreal uses three different Ubiquiti radios: NanoBeam NBE-5AC-16, NanoBeam NBE-M5-16 and Rocket M5. The M5 radios communicate via the 802.11n protocol, while the NanoBeam AC device operates over 802.11ac network protocol. A brief comparison of devices can be seen in Table 3. For more detailed device information please see the corresponding datasheets in Appendix C.

Table 3: Specifications of deployed Wi-Fi radios

	 Ubiquiti NanoBeam NBE-M5-16	 Ubiquiti NanoBeam NBE-5AC-16	 Ubiquiti Rocket M5
Frequency	5 GHz, 802.11n	5 GHz, 802.11ac	5 GHz, 802.11n
Wireless Throughput	150+ Mbits/s	450+ Mbits/s	150+ Mbits/s
Range	10+ km	10+ km	
LAN speed	100 Mbits/s	1 Gbit/s	100 Mbits/s
Deployed Locations	ONC3, ONH4, SUH28, MSH6, PKH5	SC307, SCH10, SCH6	SE5

The NanoBeam NBE-M5-16 is equipped with a 100 Mbits/s LAN port and has available services including SNMP capabilities, a web server, SSH, telnet, and NTP time synchronization.

The NanoBeam NBE-5AC-16 is a newer generation of the same radio type. It features some upgrades including a 1 Gbit/s LAN port and operates on a newer set of firmware. It has different configuration modes for PTP and PTMP and is capable of greater channel width in PTP mode.

The Rocket M5 radio also uses a 100 Mbits/s LAN port and has the same set of available features as the NanoBeam models.

Configuration can be accomplished through SSH connection, via webserver, or using Ubiquiti AirControl central management server.

2.2.1.1.2 ZigBee Radios

ZigBee is a wireless communication technology that is designed for low data rate transfer and functions on the IEEE 802.15.4 protocol. It requires little power and is designed on reliability [3].

Radios following the ZigBee protocol are designated by connector and end point. The connector attaches to the main network and each end point can connect to a device.

Table 4: Specifications of deployed ZigBee Radios

	 Digi Xbee 232 Adapter S1 Pro	 Digi Xbee-Pro 900HP RF Modem
RF Data Rate	250 kbits/s	250 kbits/s
Indoor/Urban Range	300 feet	1000 feet
Outdoor/RF Line-of-Sight Range	Up to 1 mile	Up to 28 mile
Deployed Locations	SUH28	SUH28

In deployment, ZigBee is used to maintain a serial connection over wireless. The product used in the test-system is the Digi Xbee 232 Adapter S1 Pro. This device has a DB-9M connector to attach to serial devices and contains the Digi XBee-Pro radio module which has an RF data rate of 250 kbits/s. Basic information is pictured above in Table 4 and more detail can be seen in Appendix C.

The device can be configured using the frbee Digi XCTU software.

2.2.1.2 Wired

Wired communication is capable of much faster speeds than wireless connectivity. Physical cable is used to connect devices and a device must connect to a node in the network.

2.2.1.2.1 Ethernet and Optical Fiber

Ethernet and Optical Fiber cables are used to wire the network structure together. Fiber Optic offers much faster communication speeds and throughput than Ethernet.

2.2.1.2.2 Serial

Some of the deployed devices require a serial communication line. RS232 was used as a connection medium for ZigBee radios, traffic radars, and the RFID reader. RS232 is composed of three lines – transmit, receive, and ground.

To connect serial devices to the rest of IP designated network, Serial-to-Ethernet adapters are used. These adapters copy the serial data to an IP packet within the device itself. Three models are deployed: USR IOT USR N-520, Lantronix ED2100002-01, and Perle IOLAN SDS2 W.

Table 5: Specifications of deployed Serial-to-Ethernet adapters

	 USR IOT USR N-520	 Lantronix ED2100002-01	 Perle IOLAN SDS2 W
Serial Capabilities	RS232/RS485/RS422	RS232/RS485/RS422	RS232/RS485/RS422
Protocols	HTTP, UDP broadcast	HTTP, HTTPS, FTP, Telnet, SSH, SNMP	HTTP, HTTPS, Telnet, SSH
Deployed Locations	PKH5, SUH28, SCH6,	MSH6	ONH4

Table 5 provides a list of capabilities for each adapter type and in which locations they are deployed. For more detailed device information please refer to the datasheets in Appendix C.

2.2.1.3 Bridging Networks/Mediums

Virtual Private Network Tunnels and Network Address Translation are used to bridge networks together in the smart city prototype.

2.2.1.3.1 Virtual Private Network Tunnel

A virtual private network tunnel is designed to bridge two private networks by connecting them across the public network. Cisco Meraki MX84 is the product used to create such a tunnel in the Montreal network deployment. Table 6 provides some basic specifications and to see more detailed information see Appendix C.

Table 6: Specifications of deployed VPN tunnel devices

	 Cisco Meraki MX84
Stateful Firewall Throughput	500 Mbits/s
Advanced Security Throughput	320 Mbits/s
Recommended Maximum Users	200
Maximum concurrent VPN tunnels	100

2.2.1.3.2 Network Address Translation

Network address translation is a method of changing the destination of an IP packet by changing its heading contents. This process is performed by routing devices and can be used to hide IP address information from outside or to bridge access to devices over various networks [4].

In the Montreal deployment, network address translation is set up to enable a communication pathway from the public network to several specific devices. An example can be seen in Figure 4, where port 80 on the public IP address maps to a camera on the IoT Network.

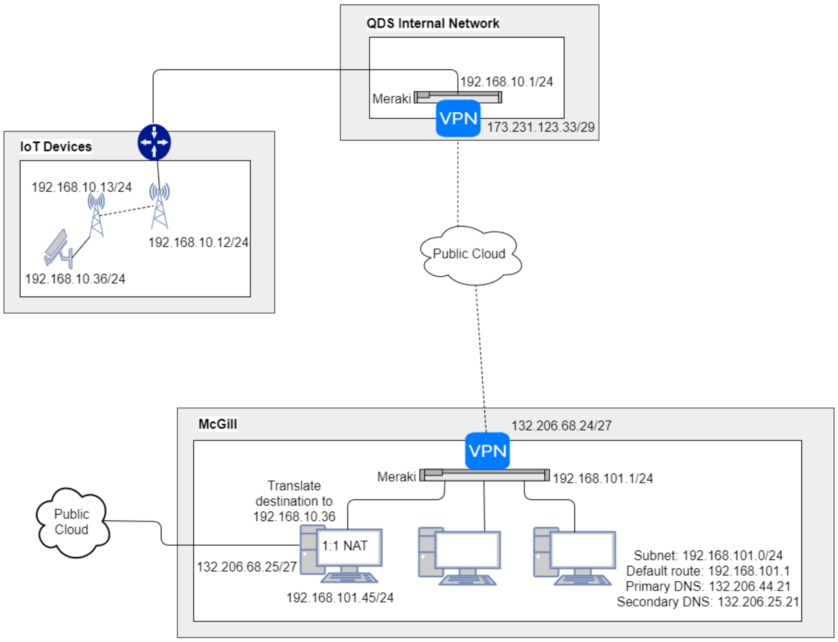


Figure 4: Diagram of NAT setup

2.2.2 Sensing

2.2.2.1 Cameras

Cameras were installed around the Quartier des Spectacles area for data processing and analytics. Four camera models are deployed Axis Q6128-E, HikVision DS-2DF6236-AEL, HikVision DS-2CD4585F-IZH, and Panasonic WV-SW598A. Table 7 details the four camera models below and more specifications can be found in Appendix C.

Table 7: Specifications of deployed cameras

	 Axis Q6128-E	 HikVision DS-2DF6236-AEL	 HikVision DS-2CD4585F-IZH	 Panasonic WV-SW598A
Max Resolution	3840x2160	1920x1080	4096x2160	1920x1080
PTZ	Yes	Yes	No zoom	Yes
Protocols	IPv4/v6, HTTP, HTTPSa , SSL/TLSa , QoS Layer 3 DiffServ, FTP,	IPv4/IPv6, HTTP, HTTPS, 802.1X, QoS, FTP, SMTP, UPnP, SNMP, DNS,	TCP/IP, ICMP, HTTP, HTTPS, FTP, DHCP, DNS, DDNS, RTP,	IPv6: TCP/IP, UDP/IP, HTTP, HTTPS, RTP, FTP, SMTP, DNS, NTP,

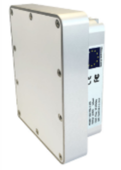
	CIFS/SMB, SMTP, Bonjour, UPnPTM, SNMP v1/v2c/v3 (MIB-II), DNS, DynDNS, NTP, RTSP, RTP, SFTP, TCP, UDP, IGMP, RTCP, ICMP, DHCP, ARP, SOCKS, SSH, NTCIP	DDNS, NTP, RTSP, RTP, TCP, UDP, IGMP, ICMP, DHCP, PPPoE	RTSP, RTCP, PPPoE, NTP, UPnP, SMTP, SNMP, IGMP,802.1X, QoS, IPv6, Bonjour	SNMP, DHCPv6 , ICMP, ARP IPv4: TCP/IP, UDP/IP, HTTP, HTTPS, RTSP, RTP, RTP/RTCP,FTP, SMTP, DHCP, DNS, DDNS, NTP, SNMP, UPnP , IGMP , ICMP , ARP
API	ONVIF S/G, CGI	ONVIF, CGI	ONVIF, CGI	ONVIF S/G, CGI
Other		36x optical zoom, optical defog	People counting, IR	30x optical zoom, waterproof, super dynamic
Deployed Locations	PKH5, SUH28	MSH6, SCH6	ONH4	SCH10

2.2.2.2 *Radars*

Radars are used to perform traffic monitoring and statistics measurements. They function by transmitting a radio signal and measuring the amount of time before the signal returns after hitting an object.

The traffic radar deployed for this Montreal IoT smart city project was the Geolux RSS-2-300 T Speed Sensor. It can communicate over the RS-232, RS-485, CAN, or Alarm open-drain outputs interfaces and can detect vehicles up to 400m away. The radar's capabilities are summarized in Table 8 and more details can be found in Appendix C.

Table 8: Specifications of deployed traffic radars

	 Geolux RSS-2-300 T
Protocols	RS-232, RS-485, CAN, Alarm open-drain outputs
Max Detection Range	400m
Measurement Precision	+/- 1 km/h
Measurement Range	5 km/h to 336 km/h
Deployed Locations	ONH4, SUH28, MSH6, PKH5, SCH6

2.2.2.3 *RFID*

RFID technology has many potential applications in a smart city. In a passive RFID system, an RFID reader outputs a signal through an antenna. If an RFID transponder tag picks up the signal, it uses the energy to turn on, run authentication processing, and transmit an outgoing signal [5].

Transponder tags can have a variety of different properties. Battery assisted tags extend range due to greater supplied power, extra user memory allows for more data storage on a tag, and some tags include hardware to generate encrypted data transfer.

Currently, passive tags are generally Class1 Gen2. This is an EPC standardization and specifies the general structure of an RFID tag. Class1 Gen2 implements an access password and kill password within the tag. The access password can be used to lock memory and render it un-writeable, and to prevent reading by readers that do not have the correct access password. If the tag kill password is sent by the reader, it will disable the tag permanently [6].

Two versions of Class1 Gen2 tags exist, V1 and V2. V1 is the current standard and is much cheaper to produce, but has inherent security flaws [6, 7]. V2 expands on V1's security principles by implementing file management, and improved security through AES encryption [8].

Table 9: Specifications of deployed RFID reader

	 FEIGISC.LRU1002 (Pre-2017 model)	 BW-BLEG-WME
Protocols	TCP/IP, RS232, USB	TCP/IP, Ethernet
Tag Support	Class1 Gen2 V1, Class1 Gen2 V2	Read/write Beacon, iBeacon Tag
Max Number of Antennas	4	1 (integrated ceramic antenna)
Max Antenna Output	2 W	2 mW
Deployed Locations	BL1200	BL1200

The RFID reader deployed for field testing is a Pre-2017 model FEIG ISC.LRU1002. This device is capable of reading and communicating with EPC Class1 Gen2 V1 tags as well as EPC Class1 Gen2 V2 which features secure encrypted communication. The device can supply an output power of 2 W to each of four possible antennas. A summary of the reader's capabilities can be seen in Table 9 and the more detailed datasheet can be found in Appendix C.

2.3 Mobile Device Models

This section will detail the device models mounted on mobile vehicles. These include both a winter salt truck and a road surface and tree inspection vehicle.

2.3.1 Salt Truck

Salt trucks are important for Montreal winters as they disperse salt on the roads to de-ice and also plow away snow from streets. The lab mounted additional sensors to the truck to improve its efficiency and monitoring abilities. The truck was retrofitted with an ambient and road temperature sensor to measure driving conditions, an ultrasonic level sensor for measuring salt level in the container, and an LTE gateway and various adapters for communication.

The connection diagram of mobile sensor devices is presented in Figure 5. It shows sensor modules and the communication mediums and adapters required for the system to function. The LTE gateway and all the adapters are stored together inside one box.

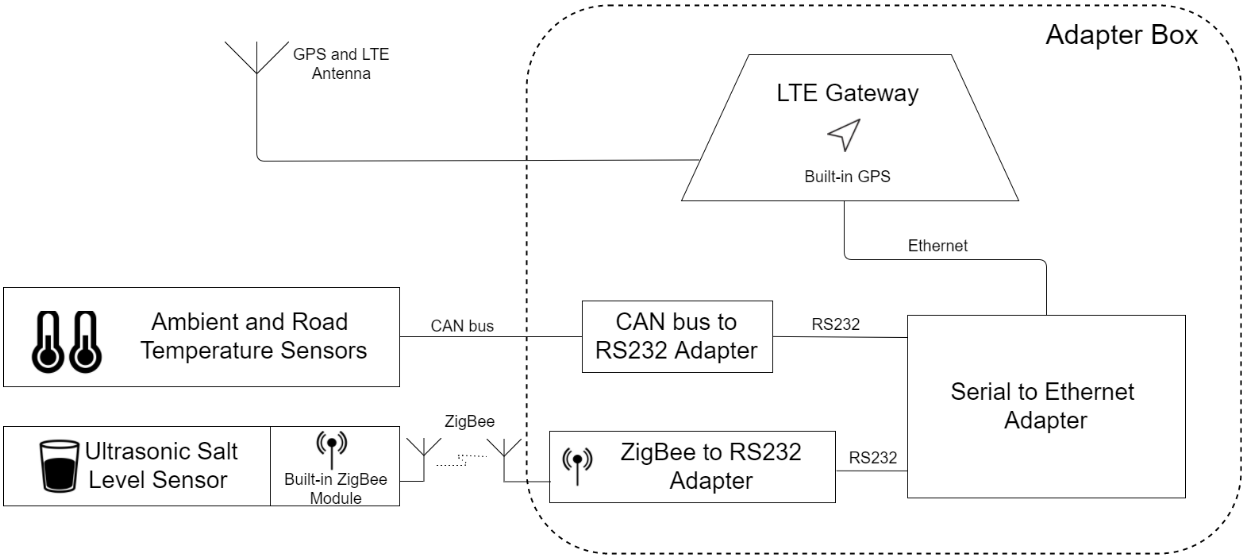


Figure 5: Mobile Sensors connection diagram

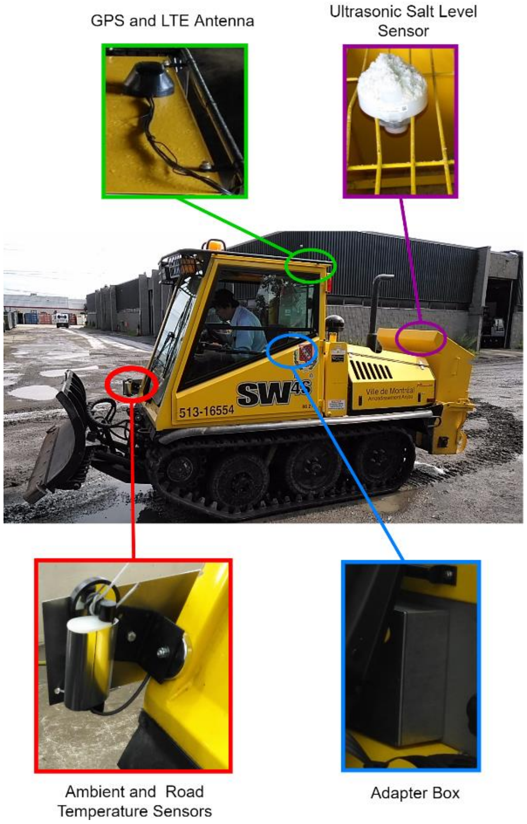

GPS and LTE Antenna
Ultrasonic Salt Level Sensor
Ambient and Road Temperature Sensors
Adapter Box

Figure 6: Pictures presenting mounted sensor installation locations on LTE-connected truck

The physical installation locations are shown in Figure 6. The temperature sensors are installed near the front of the truck with a clear line of sight for the road temperature sensor. The adapter box is installed inside the cabin behind the seat for protection from the elements, and the salt level sensor is placed on a grate overlooking the salt container. Mounted to the top of the truck roof is the GPS and LTE antenna to collect and send data.

2.3.1.1 LTE Gateway

An LTE gateway needed to be mounted on the salt truck seen in Figure 6 to allow mobile data transfer. The device deployed as the gateway is a Sierra Wireless Airlink GX450. It is capable of connectivity through both 802.11b/g/n Wi-Fi and LTE. It features RS-232 serial, USB, Ethernet, and GPS antenna connectors. Table 10 highlights device information and further specifications can be found in Appendix C.

Table 10: Table of Sierra Wireless Airlink GX450 specifications

	 Sierra Wireless Airlink GX450
Protocols	Ethernet, RS-232, digital I/O, USB, Cellular, Wi-Fi, SNMPv1, SMS commands, Telnet, SSH
Security	Up to 5 VPN tunnels, WEP, WPA-PSK, WPA2-PSK
Operating Temperature	-30C to 70C
Storage Temperature	-40C to 85C
Ruggedness	Military Spec MIL-STD-810G conformance to shock, vibration, thermal shock, and humidity

2.3.1.2 Temperature Sensor

Temperature sensors were attached on the salt truck to measure ambient and road temperature. The temperature sensor installed is the CVG RoadWatch SS along with the CVG RoadWatch SS RS-232 adapter. It is capable of accuracy within 2F to 6F depending on temperature range and medium measured. It features a 15-degree field of view and 1/10 second response time. Device specifications are summarized in Table 11 and more information can be found in Appendix C.

Table 11: Massa RoadWatch SS specifications

	 CVG Roadwatch SS

Type	Passive Infrared
Weight	11 oz
Temperature Range	Road surface: -40F to 150F Road accuracy: Within 2F (23F to 41F ambient) Within 6F (-40F to 23F, 41F to 150F) Air: -40F to 131F Air accuracy: Within 2F (-40F to 131 F)
Response Time	1/10 seconds
Field of View	15 degrees

2.3.1.3 Level Sensor

The Massa Model M3 wireless tank level sensor was installed to enable salt-level monitoring within the truck container. The device is an ultrasonic level sensor and has a built-in ZigBee module to send information wirelessly. It is reprogrammable and has an ultrasonic range response time of 150ms to 500ms. Table 11 presents basic specifications and more can be found in Appendix C.

Table 12: Massa M3 Level Sensor specifications

	 Massa M3 Wireless Tank Level Sensor
Operating temperature	-30C to 65C
Ultrasonic range response time	150ms to 500ms
Data acquisition interval	Programmable 10s to 194 days
Battery life	3 years
Power	3 Lithium Energizer AA batteries

2.4 Road Surface and Tree Inspection Vehicle

The road surface and tree inspection vehicle will drive around Montreal collecting video of roads and trees along the road to notify the city of pothole locations and tree maintenance requirements. The van is fitted with two video recording devices with information stored onto a hard drive. When in a Montreal Wi-Fi area, the contents of the hard drive are uploaded to a data processing center. The interconnectivity of devices can be seen in the block diagram in Figure *7*.

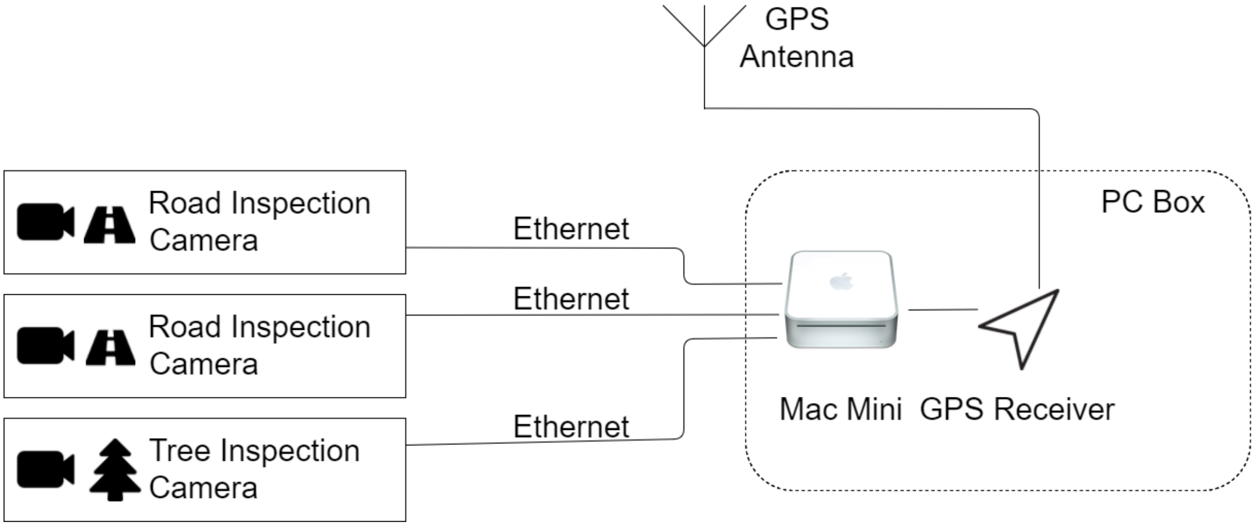


Figure 7: Block diagram of road surface and tree inspection vehicle setup

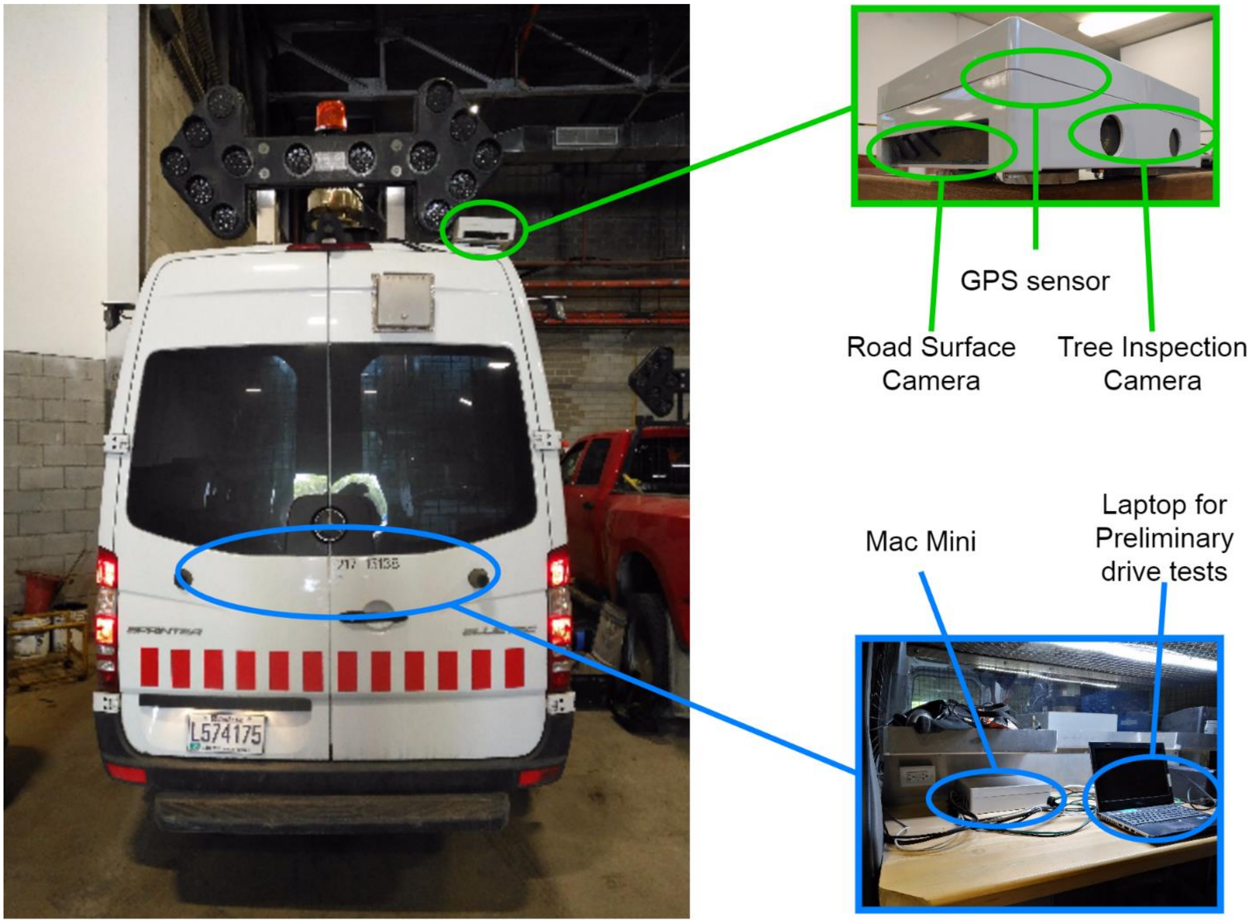


Figure 8: Detailed installation setup for road and tree inspection vehicle

Physical installation setup can be seen in Figure 8. The road surface and the tree inspection cameras are installed into one housing and mounted at the back of the vehicle for a clear view. The road surface camera is looking downwards to the road at the back of the vehicle and the tree inspection camera is facing sideway to the right side of the vehicle for tree inspection. These devices connects to a Mac Mini inside the truck which will stored videos from the cameras and also store the GPS logs from the GPS sensor. The videos are timestamped at the time of recording so that latter processing can match the location of sick trees or pot holes with the location information from the GPS logs.

2.4.1 Road Surface Cameras

Two FLIR Grasshopper3 were mounted on the vehicle to record road surfaces. It features a 2.3 MP camera with 1920x1200 resolution at a frame rate of 48 frames per second, operates between 0C and 50C, and uses the Sony IMX174 sensor.

Table 13: Road surface camera information

	FLIR GrassHopper3 GS3-PGE-23S6C-C
Resolution	1920x1200
Frame Rate	48 FPS
Exposure Range	0.017ms to 32s
Operating Temperature Range	0C to 50C
Dimensions	44mm x 29mm x 58mm

2.4.2 Tree Inspection Cameras

A 3D vision stereoscopic camera is installed to provide tree inspection recordings. The device used is the FLIR BumbleBee2 BB2-08S2C-38. It is 0.8 MP, supports a resolution of 1032x776 at 20 frames per second, and uses the Sony ICX204 sensor. More information can be seen in Table 14 and the datasheet in Appendix C.

Table 14: Basic information regarding tree inspection camera

	FLIR BumbleBee2 BB2-08S2C-38
Resolution	1032x776
Frame Rate	20 FPS
Aperture	f/2.0
Exposure Range	0.03ms to 66.63ms
Operating Temperature Range	0C to 45C
Operating Humidity	20%-80% (No condensation)
Dimensions	157mm x 36mm x 47.4mm

2.5 Time Synchronization

In a full-scale IoT smart city deployment, it can be expected that there will be tens of thousands of devices sending their information to data processing centers. Unfortunately, communication latencies are highly variable and there is no accurate way of predicting how long it will take for a data packet to reach its target destination.

Time stamps are an important part of correlating data and maintaining records, and may be critical in time sensitive implementations. Due to data communication latencies, it is much more accurate to generate a time stamp at the device when the data is generated. Still, time must somehow be synchronized across the thousands of devices to maintain accurate time relevancy. The following section will detail current time synchronization protocols, ongoing research, and recommendations for a developing smart city platform.

There are three primary time synchronization protocols employed today: Network Time Protocol, Precision Time Protocol, and Global Positioning Satellite.

Network Time Protocol

Network Time Protocol (NTP) is the most commonly used time synchronization scheme in commercial applications. Thus, most devices are NTP compatible and ready for easy set up. NTP generally functions on the software level and uses a two-way message-response communication to synchronize. Timestamps are appended to a message originating from the client, and the times between these timestamps are used to accurately estimate latency and clock offset between the client and server [9]. Note that the accuracy of the protocol greatly increases when routing is the same bi-directionally.

Specifically, four timestamps are compared to synchronize time: Timestamp message sent to server T_0, timestamp of message received by server T_1, timestamp of message reply sent from server to client T_2, timestamp of message reply received by client T_3. Using these times, two-way latency can be calculated as $(T_3 – T_0) - (T_2 – T_1)$ and clock offset between client and server can be calculated as $(T_1 – T_0)/2 + (T_2 – T_3)/2$ [9].

The NTP protocol generally relies on several external servers to determine the proper time for the network server. The timestamps from each of these external servers are compared algorithmically, with the outliers removed, to determine the absolute time [9].

Precision Time Protocol

Precision Time Protocol (PTP) functions using the same calibration methodology as the Network Time Protocol. The difference arises in implementation standards. As NTP generally functions on the software level, extra latencies and variances result due to operating system processing. PTP overcomes this by keeping everything at the hardware level. Hardware timestamps are generated and sent rather than software timestamps [10]. This significantly improves time relevant accuracy across a system, but requires additional hardware capable of PTP.

Global Positioning Satellite

Table 15: Comparison of accuracy of various time synchronization methods [11, 12]

Protocol	Synchronization Accuracy
NTP	50-100 ms
PTP	20-100 ns
GPS	10-20-100+ ns Highly dependent on GPS receiver

Global Positioning Satellite (GPS) time synchronization is also a frequently used synchronization method. It makes use of GPS satellites in orbit around the Earth. A GPS receiver is used to compare signals broadcasted by several satellites. By comparing time differences between signals, and using triangulation position calculations, time is determined [11].

The GPS synchronization method requires GPS receivers unobstructed from receiving the broadcasted GPS signals.

Table 15 shows a comparison of the time synchronization accuracy achievable by each time synchronization method.

Recommendations

A suggested time-synchronized system is one which has several GPS receiver servers to determine an accurate absolute time. One of these servers can be the starting point for the PTP setup. PTP clocks with NTP ability should be employed routinely, especially at major nodes along the system. Devices incapable of using the PTP system should get NTP time from the nearest PTP clock. This will synchronize the system as accurately as possible.

NTP time synchronization is currently enabled on all IP devices and is synchronized to the 192.168.101.45 management server.

2.6 IP Addressing

There are two generations of IP addressing: IPv4 and IPv6.

IPv4 addressing relies on four 8-bit sections. An example is 192.168.101.45. It works in a hierarchical manner, with the first two sections designating the network, the third determining a sub-network, and the last selecting the host on the network and sub-network [13].

IPv6 was developed due to the large influx of devices connected to the internet. Moving into an IoT world, IPv4, the de-facto standard, will simply be incapable of addressing all devices effectively and thus IPv6 expands to allow for more devices. It uses a total of 128 bits to designate an address rather than the 32 in IPv4. The 128 bits are broken down into 8 sections [13].

IPv4 is currently supported by all IP enabled devices, while not all are compatible with IPv6 addressing.

Currently, all the devices in the test-deployment of IoT smart city are addressed using IPv4.

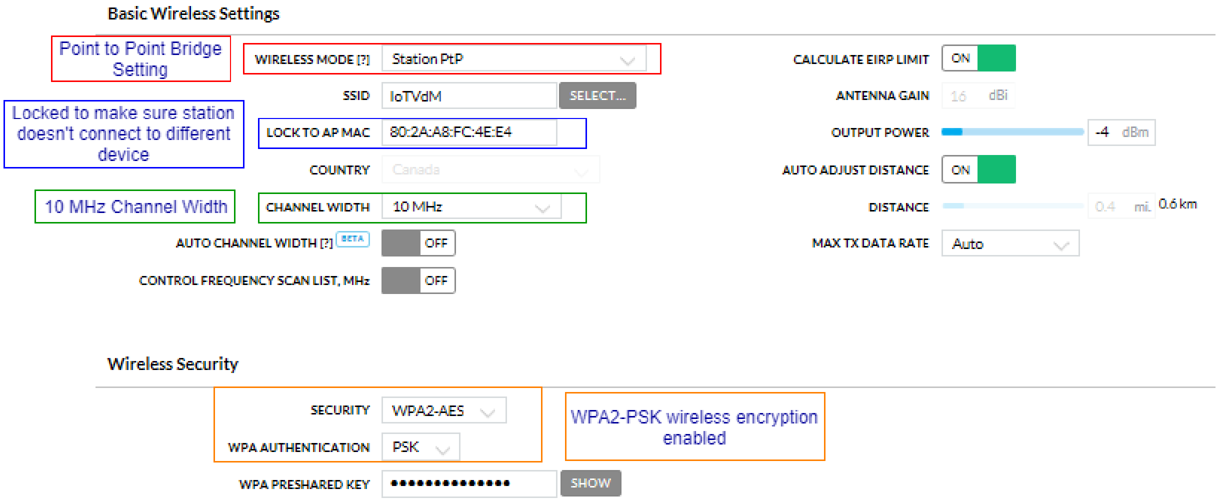
3 Configurations

This section will detail the configuration setup of devices and how certain infrastructures such as network address translation are set up.

3.1 Wi-Fi Radios

As the Wi-Fi radios were deployed in a busy section of downtown Montreal, they were susceptible to a large amount of interference noise. To provide the maximum available throughput, the radio channel frequency was set to the one featuring the lowest interference level using the spectrum analyzer. In an effort to maintain throughput stability, a minimum channel width of 10 MHz was selected. Output power is also set to a maximal level of -4 dBm to maintain as strong of a signal connection as possible.

The Ubiquiti Wi-Fi radios are enabled in such a manner as to maintain high levels of security, but to allow easy experimentation with settings and tools. Each ST was locked to its accompanying AP to allow it to maintain a connection to the proper source. The wireless communication channel between the two devices was also encrypted via WPA2-PSK. Each of these configuration setting can be seen below in Figure 9 which shows a screenshot of the webserver wireless settings.



Basic Wireless Settings

Point to Point Bridge Setting

WIRELESS MODE [?] Station PtP

SSID IoTVdM

Locked to make sure station doesn't connect to different device

LOCK TO AP MAC 80:2A:A8:FC:4E:E4

COUNTRY Canada

10 MHz Channel Width

CHANNEL WIDTH 10 MHz

AUTO CHANNEL WIDTH [?] BETA OFF

CONTROL FREQUENCY SCAN LIST, MHz OFF

CALCULATE EIRP LIMIT ON

ANTENNA GAIN 16 dBi

OUTPUT POWER -4 dBm

AUTO ADJUST DISTANCE ON

DISTANCE 0.4 mi. 0.6 km

MAX TX DATA RATE Auto

Wireless Security

SECURITY WPA2-AES

WPA AUTHENTICATION PSK

WPA2-PSK wireless encryption enabled

WPA PRESHARED KEY ••••••••••••••

Figure 9: Webserver configuration page for wireless settings

Several services were turned on the radios: Ping Watchdog, SNMP Agent, Web Server, SSH Server, and NTP client. Ping Watchdog was enabled to limit radio disconnections due to LAN port faults. When an AP device loses connection to the main system gateway or if a ST loses connection to the connected device for a successive period of 6 minutes, it is set to restart under Ping Watchdog. SNMP is enabled to allow for SNMP management protocol testing. The Web Server provided an intuitive interface for changing configurations and running tests in an experimental setting. The SSH Server was enabled to provide SSH access to the central management platform. The NTP client was needed to keep time synchronized effectively across all devices.

3.2 Cameras

To connect to the cameras from the radios, Ethernet cable is used. Each camera is enabled to communicate via http and each has its own unique address. Video resolution is set to 1280x720 to provide clear video, yet limit the amount of data transmission throughput.

ONVIF is also enabled on each of the cameras to provide a standard control method.

3.3 RFID

The object of the RFID reader is to help maintain inventory records of whether certain items are in storage or in-use.

For UHF RFID setup, inventory items are marked with RFID transponder tags and carried in and out through a choke point, where the reader is set up to scan tags which pass through. The reader connects to the network via its RS232 port. The RS232 port communicates over a ZigBee radio channel to an RS232 to Ethernet adapter. The RFID reader is configured in scan mode, which outputs programmed transponder data through the RS232 port without external signals from a host. The port on the Ethernet adapter is monitored to collect data and update the database when a transponder signal is sent.

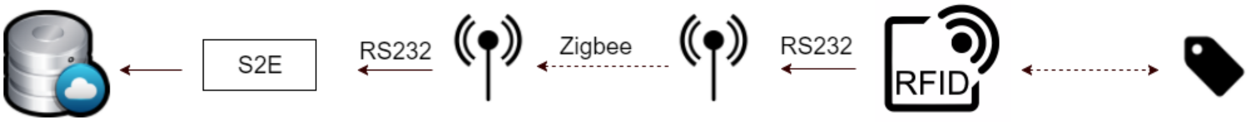


Figure 10: Diagram showing data transmission setup of RFID

The current deployment setup involves use of Class 1 Generation 2 tags which feature no encryption mechanism. Data can be minimally secured behind the access password, but this is not a strong system, as information is sent via standard text. The access password is, however, used to prevent accidental readings of non-authenticated tags. Both the reader's authentication password and the transponder's access password must match for the tag information to be read. Once the reader deciphers the tag information, it pushes tag data through the RS232 port. It pushes the same tag's information only once every 30 minutes.

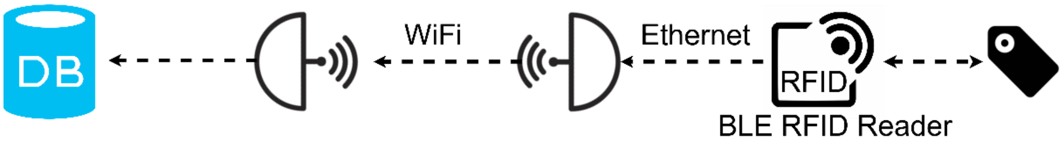
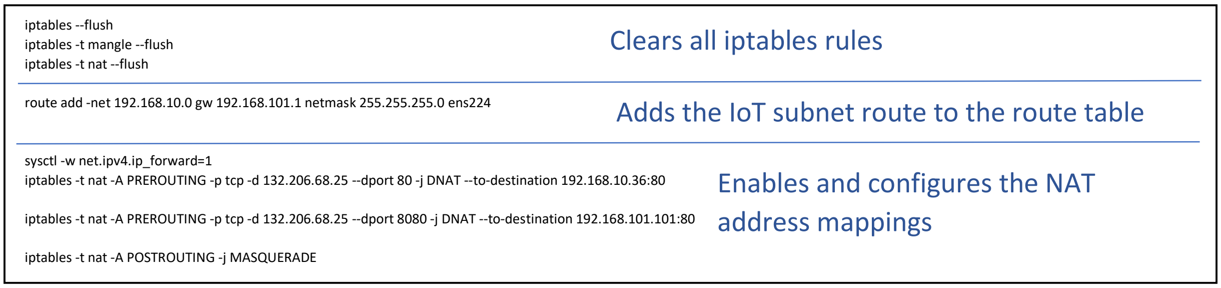


Figure 11: Data transmission setup of BLE RFID reader device.

Similarly setup is used for BLE RFID as shown in Figure 11. However, since BLE RFID reader is equipped with an Ethernet port, it can connect directly to the existing WiFi infrastructure.

3.4 Network Address Translation



```
iptables --flush
iptables -t mangle --flush
iptables -t nat --flush
```

Clears all iptables rules

```
route add -net 192.168.10.0 gw 192.168.101.1 netmask 255.255.255.0 ens224
```

Adds the IoT subnet route to the route table

```
sysctl -w net.ipv4.ip_forward=1
iptables -t nat -A PREROUTING -p tcp -d 132.206.68.25 --dport 80 -j DNAT --to-destination 192.168.10.36:80

iptables -t nat -A PREROUTING -p tcp -d 132.206.68.25 --dport 8080 -j DNAT --to-destination 192.168.101.101:80

iptables -t nat -A POSTROUTING -j MASQUERADE
```

Enables and configures the NAT address mappings

Figure 12: Script that enables and maps network address translation

To connect several devices to public network access, a virtual machine CentOS server is set up as a routing device. The virtual machine has access to several network interface cards with one card connected to the public network with IP address 132.206.68.25 and another referenced on the IoT server subnet. To allow beyond a 1:1 mapping for the public IP address, different ports are translated to different addresses. For example, when a packet is sent to port 80, it translates the address to the CAM_SUH28_1 camera's 192.168.10.36 address, but port 8080 map to IP address 192.168.101.101.

Setting up network address translation was accomplished by use of iptables and route. A service was created to run the script seen in Figure 12 at startup. The script clears all iptable rules, adds the route rule for the 192.168.10.x subnet and initializes all the network address translation settings.

3.5 Time Synchronization

Time synchronization across the network was set using the 192.168.101.45 management server. This centOS server was set to run ntpd, a network time protocol daemon. The server synchronizes itself with 4 public time servers to determine its own relevant time.

4 Test Results

This section will detail tests and results completed relating to the network structure. The section will provide information on device throughput and reliability.

4.1.1 Throughput

Tests were conducted to determine throughput capabilities in an ideal lab scenario as compared to when deployed in a city. These tests were conducted for Wi-Fi and ZigBee radios.

4.1.1.1 Wi-Fi Radios

Laboratory experiments on Ubiquiti NanoBeam NBE-M5-16 and NanoBeam NBE-5AC-16 throughput rates were performed prior to deployment for future comparison and reference. Each radio type was configured and tested in a PTP and PTMP [3 node] setting to compare maximum throughputs with different setups. For the AC radios, both the PTP and PTMP mode settings were tested in the physical PTP setup. As discussed previously, the M5 radios do not have the option to select a PTP mode.

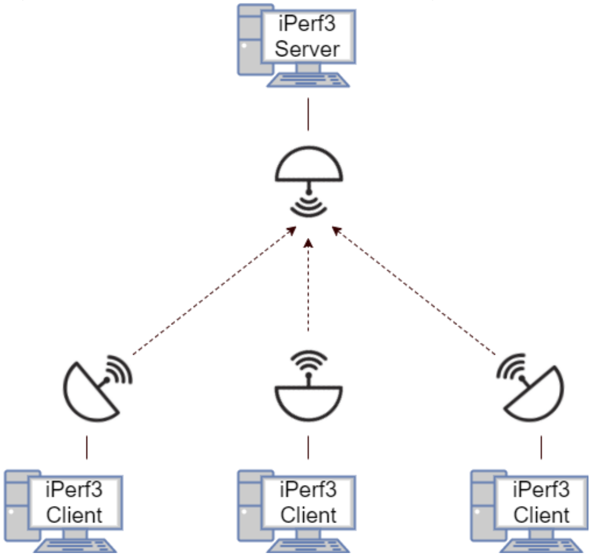


Figure 13: Diagram of the PTMP [3 node] setup

Initial experimentation was performed using the iperf3 network performance management tool. This tool allows one device, the client, to generate IP traffic and direct it to another listening device, the server. To generate maximum traffic, the UDP protocol was used as it does not rely on a two-way communication connection. As this test was attempting to pump through as many packets as quickly as possible, this resulted in a large packet loss percentage and is likely not representative of a real-life usage scenario.

To simulate an idealized deployment scenario, one radio was initialized as an AP and was connected to the server computer, while the rest designated as ST radios and connected to client computers. In deployment, the AP would be connected to the rest of the network and the ST to a device such as a camera. The radios serve as the communication bridge, sending sensor data wirelessly to the AP and, as a result, the rest of the network.

Figure 13 shows the described PTMP [3 node] setup with iperf3 running in server mode on the desktop connected to the AP radio and desktops running client mode connected to ST radios.

The following commands were used by each client in a given set up.

```
iperf3 -c <server address> -u -p <port #> -P <# of parallel streams> -b <packet size>
iperf3 -c 192.168.101.45   -u -p 5201     -P 20                       -b 60M
```

The following command was used to set up the listening server.

```
iperf3 -s -p <port number>
iperf3 -s -p 5201
```

Each radio was set to a 40 MHz channel width, without security enabled and device MAC addresses were not locked together.

The following graph in Figure 14 shows the results of the iperf3 tests and accompanying data presented by the Ubiquiti AirOS webserver console. The generated results were similar.

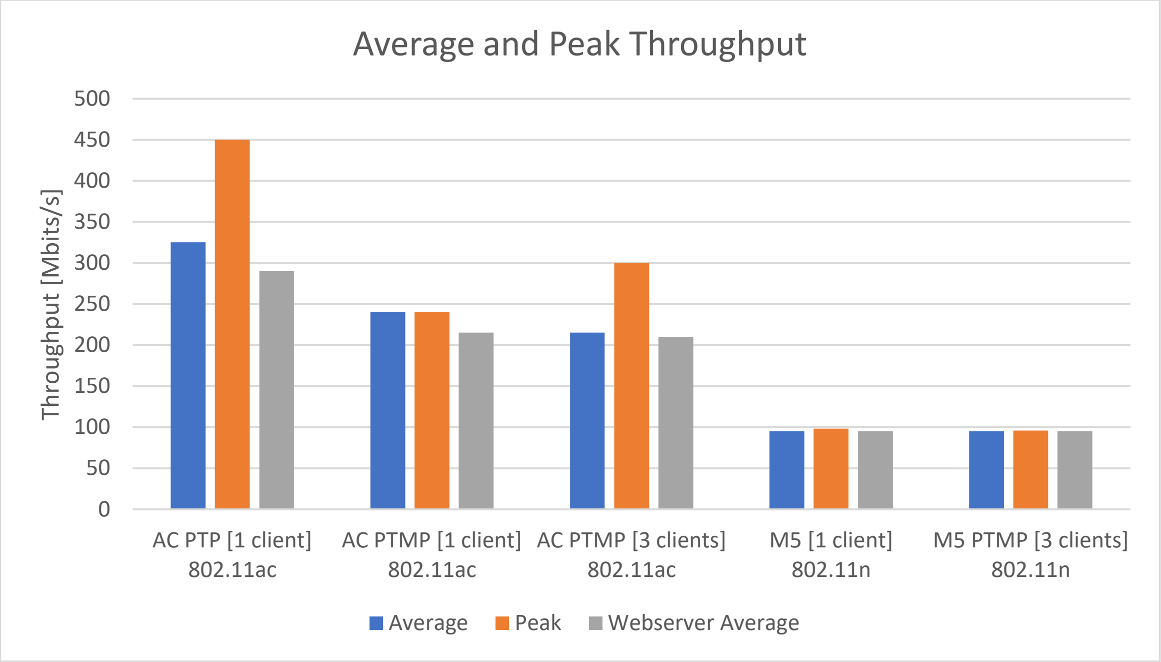


Figure 14: Chart of throughput results

The iperf3 test verified and put some manufacturer advertised data rates into perspective. Ubiquiti advertises 450+ Mbits/s throughput speeds for the NanoBeam NBE-AC-16 devices, and while a peak of 450 Mbit/s was noted in PTP mode, it was not consistent. The average hovered at 325 Mbit/s. The data rate suffered a large performance hit when the AC radio was switched to the PTMP configuration mode setting, even while remaining in a physical PTP setup. With the configuration change the throughput dropped to an average of 240 Mbits/s. While adding an additional two nodes, the throughput dropped an additional 25 Mbits/s to 215 Mbits/s, but this was a rather insignificant drop in comparison to the switch from PTP to PTMP modes. Regardless of PTP or PTMP physical configuration, the M5 radios' average and peak throughputs remained near 100 Mbits/s, which is likely a limit restricted by the devices' 100 Mbit/s max LAN speed. Ubiquiti claims the M5 radios are capable of 150+ Mbits/s wireless transmission throughput.

After completion of the basic laboratory throughput experiments, the radios were deployed in downtown Montreal. Data was collected to determine the throughput and interference levels in an urban deployment setting.

Initially the radios were deployed with the same 40 MHz channel width frequency settings that were tested in the lab. Due to large interference sources in certain frequencies, and the 40 MHz channel, encompassing a large frequency range, some of the 802.11ac radios dropped to below 5 Mbits/s average throughput rate from 325 Mbits/s in lab. The channel width was therefore decreased to 10 MHz and centered in a frequency with the lowest interference. An example spectrum analysis from the Ubiquiti NanoBeam webserver console can be seen in Figure 15, with a 40 MHz channel width highlighted with a red line and 10 MHz highlighted with blue. While the 10 MHz width can be placed in a frequency channel with relatively little interference, this is not the case for the much larger 40 MHz channel width.

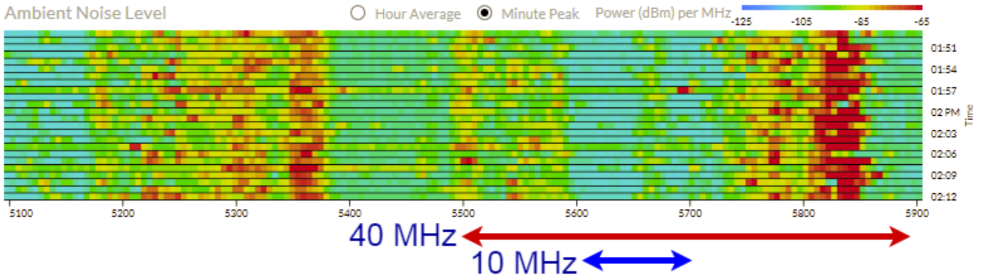


Figure 15: Spectrum analysis for a Ubiquiti NanoBeam AC radio in downtown Montreal with 40 MHz and 10 MHz widths highlighted

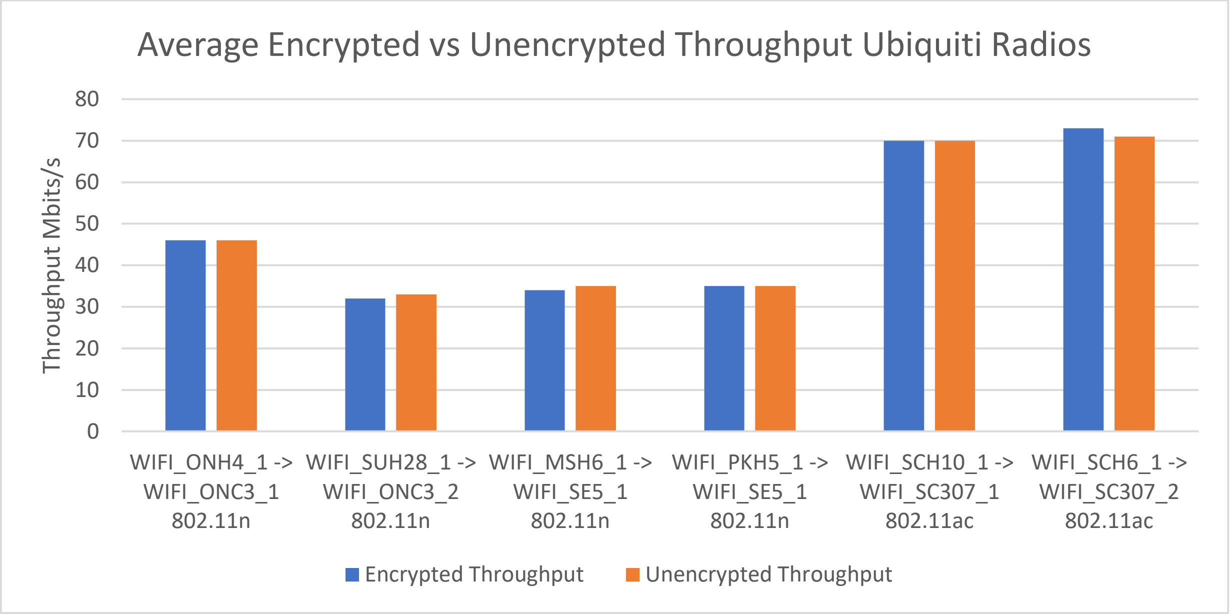


Figure 16: Comparison of Encrypted vs Non-Encrypted Deployed Throughput

After decreasing the channel widths of all deployed radios to 10 MHz, throughput tests were performed again. Tests were performed with WPA2 wireless encryption enabled and disabled. The Ubiquiti AirOS interface was used to determine interference levels and maximum speed capacity, and was verified by using the iperf and iperf3 tools installed within the Ubiquiti radios. Iperf and iperf3 were accessed via ssh terminal connection to a radio because it was not possible to connect the radios to a separate computer to run iperf3 tests. Results can be found in Figure 16.

Radio throughput was tested with encryption both enabled and disabled. No significant effect on radio throughput was found with some radios showing minimally faster or lower maximum bandwidths.

In order to provide a more accurate comparison between the laboratory experiments and the deployed radios, the laboratory results must be scaled down, as larger unimpeded channel widths will always have higher throughput levels than smaller widths. Per Shannon-Hartley Theorem, throughput increases linearly with channel width [14]. As the channel width decreased from 40 MHz to 10 MHz, the original results can be scaled down by a factor of 4 to estimate the 10 MHz laboratory throughput speeds. Due to the LAN port on the M5 radios limiting the throughput, for a more accurate comparison, the advertised rates from Ubiquiti of 150 Mbits/s are scaled down. Results are averaged for radio type in PTP mode in deployment and can be seen in Figure 17 below.

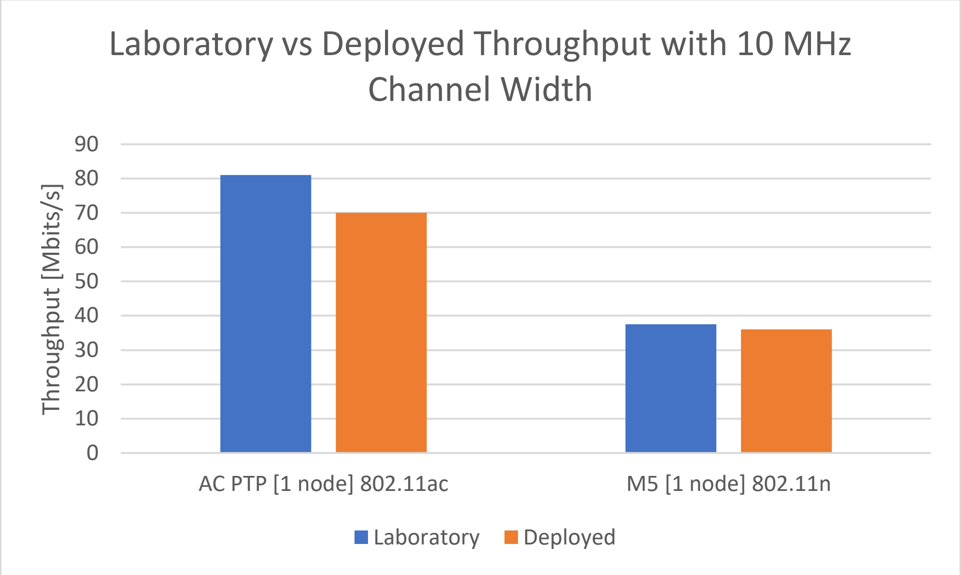


Figure 17: Figure comparing Wi-Fi radio laboratory and deployed throughputs with 10 MHz channel widths

When averaged, the throughput experienced a small decrease in deployment as compared to normalized laboratory results for 10 MHz channel width. There is some variation in the test results for the M5 802.11n radios, with the throughput between WIFI_ONC3_1 and WIFI_ONC4_1 showing greater throughput speed than the scaled down Ubiquiti advertised rates. Unfortunately, without retesting in the lab with 10 MHz bandwidth, we cannot determine an extremely accurate reference. Deployment in downtown Montreal is susceptible to too much interference for 40 MHz channel width to be used.

4.1.1.2 ZigBee Radios

The ZigBee radios were subjected to similar throughput tests to the ones conducted for Wi-Fi radios. However, different performance measurement tools needed to be employed, as the ZigBee radios only have a RS232 line and no Ethernet/IP port.

To test radio throughput performance, Digi XCTU software was used. This software can display the signal strength between a pair of radios and run a throughput test. In the throughput test menu, a payload of 230 bytes was selected, as this was the largest payload that allowed the test to function without freezing. The test was performed via loopback, in which the receiving radio's receive and transmit pins were shorted together so that it would transmit the same message back to the radio connected to the XCTU program.

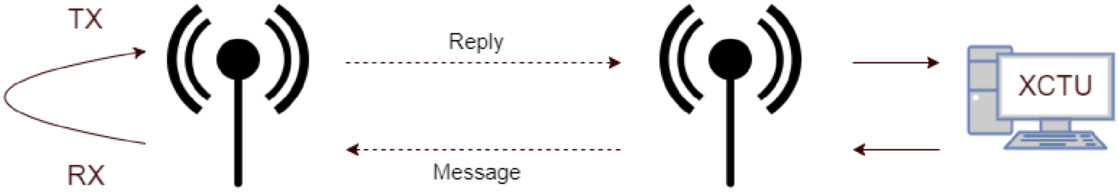


Figure 18: Diagram showing ZigBee throughput test setup

The throughput test was conducted with AES encryption enabled in which a four-byte encryption key was used as the passkey. The radios were first tested in the lab with about 1m distance separated between them and then tested in an urban environment to determine the effects of distance on throughput.

Table *16* shows the test results with the 2.4GHz radios. It is observed that there is a significant difference in throughput depending on whether the communications were encrypted or not. With distance, the throughput also drops considerably. On the radio datasheet, the Zigbee radio operates at an RF Data Rate of 250 kbits/s, however, this is the maximum bit rate over the air including all the headers and protocol overheads, and not the maximum throughput. It is also important to note that any distance larger than 50m, the 2.4GHz Zigbee radios cannot establish a stable connection, thus this 2.4GHz frequency band may not be suitable for communications in an urban environment due to the effect of signal degradation.

Table 16: Throughput test results on ZigBee Pro 2.4GHz radio

Distance	Unencrypted	Encrypted
1m	31 Kbps	9 Kbps
50m	14 Kbps	7 Kbps

The 900MHz Zigbee radios also went through similar tests. However, as seen from their specs, there is only a marginal difference in terms of throughput when the radios are configured with and without encryption; thus, only tests with encryption enabled were considered

Table 17: Throughput test results on Zigbee 900MHz radio.

Distance	Throughput (peak/average)	Received Signal Strength	Packet Loss
1m	16.4 / 16.4 Kbps	-40dBm	0%
50m	16.4 / 16.4 Kbps	-40dBm	N/A
340m	12.3 / 11.75 Kbps	-61dBm	7.6%
415m	11 / 2.5 Kbps	-59dBm	33.3%

As shown in Table 17, the 900MHz Zigbee radios can reach much further distance than the 2.4GHz radios. As further reach is more suitable for urban area, it is interested to do more extensive tests on the radios with the results of the peak/average throughput, the received signal strength and packet loss are shown in Table 17. The results illustrate that the 900MHz radio is able to establish a stable connection up to 340m with a packet loss of 7.6%. However, any distance further than will degrade the signal quality significantly due to path loss and shadowing effect, which results in significantly higher packet loss.

It is note that in the current MSCPS, the 900MHz Zigbee radios were deployed at location BL1200 with approximately 42m kink between them

According to the specs, the 900MHz Zigbee radios can achieve approximately 105Kbps with point to point unicast configuration (encryption enabled). In the mesh configuration, it can achieve up to 90Kbps with 1 hop communication and 16Kbps over 6-hop multi-hop communication. To verify these numbers, test setups are conducted to measure the unidirectional throughput (the mesh tests were not done due to the lack of Zigbee parts). In this setup, both Xbee radios were connected to the same PC. In one setup, the two radios were connected together by a coax cable with 60dB attenuator. In the second setup, the radios were connected to separate dipole antennas and communicated through the air with 5 feet distance.

Table 18: Throughput test results (unidirectional) on Zigbee 900MHz radio.

Test case	Throughput (peak/average)
60dB attenuator	90.3 / 79 Kbps
5ft wireless connection	73 / 60 Kbps

Table 18 shows the results of the tests. It is observed that the achievable throughput in the unidirectional tests with attenuated coax cable is similar to the specs of the radio. The difference may come from different setups for the test (unfortunately the manufacturer does not specify how the tests were conducted). It is also noted that the unidirectional throughput is much higher than the loopback tests. The difference may due to the protocol overhead. As a result, for actual implementation, real tests need to be conducted instead of relying on the advertised specs from the manufacturer.

The 2.4 GHz Zigbee radios were also tested in unidirectional communication for comparison purpose. As shown in Table 19, it is illustrated again that there is a huge difference in throughput between the cases of encrypted and unencrypted communications. It is also illustrated that the achievable throughput at this close distance can match to the 900MHz Zigbee radios.

Table 19: Throughput test results (unidirectional) on Zigbee 2.4 GHz radio.

Test case	Throughput (peak/average)
1m wireless connection (Unencrypted)	80 / 80 Kbps
1m wireless connection (Encrypted)	20 / 20 Kbps

4.1.2 Device Reliability

Through the central management platform described in Appendix F, Statistics are logged to determine the reliability of all deployed devices on the network. A ping test is used to check whether a device is accessible from a management server every 20 seconds. The following Figure 19 presents the average percentage of time that each of the devices were reachable by ping.

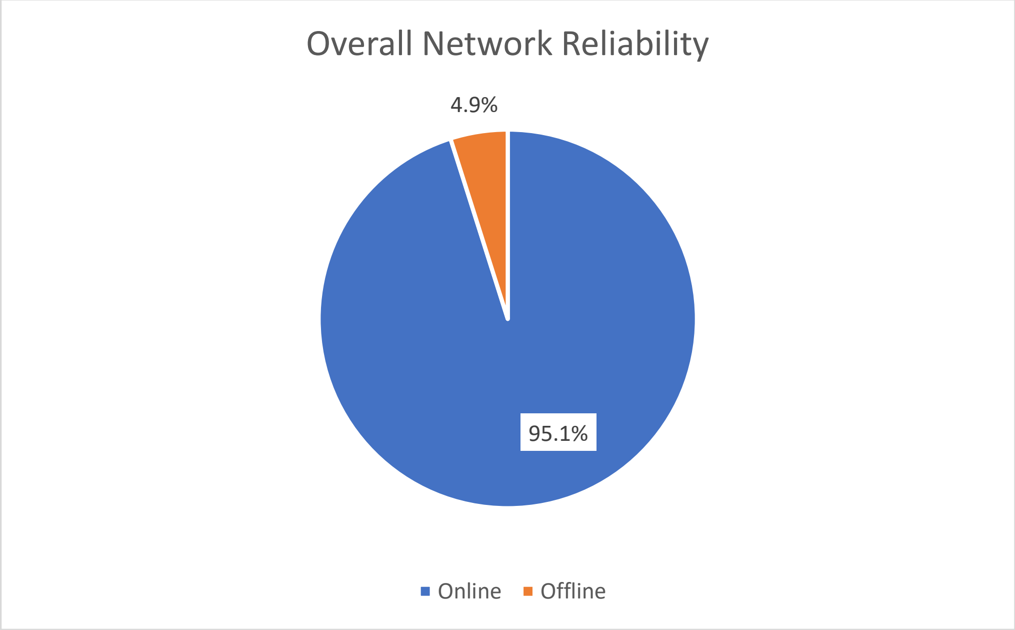


Figure 19: Device Reliability Measured by Ping

Device reliability was measured over a period of 12 days. A couple significant network issues were encountered over this time. First, there was a LAN connection issue between a station radio and its attached adapter and camera. This resulted in the camera and adapter appearing offline for 4 days. Overlapped with this 4 day period, there was also a network crash on the Quartier des Spectacles network for one day. This crash affected the before-mentioned radio and its access point, as well as another access point-station pair and their accompanying devices.

The abovementioned network disruptions can happen relatively frequently. Due to installation locations being in a shared location with festival setup teams, the prototype smart city network devices occasionally get disconnected by external parties. Some of the radios also have faulty LAN ports which provides an appearance that a connected device disconnected when the real problem is on the radio. Wi-Fi interference sometimes results in brief disconnection, but the effect of this is limited.

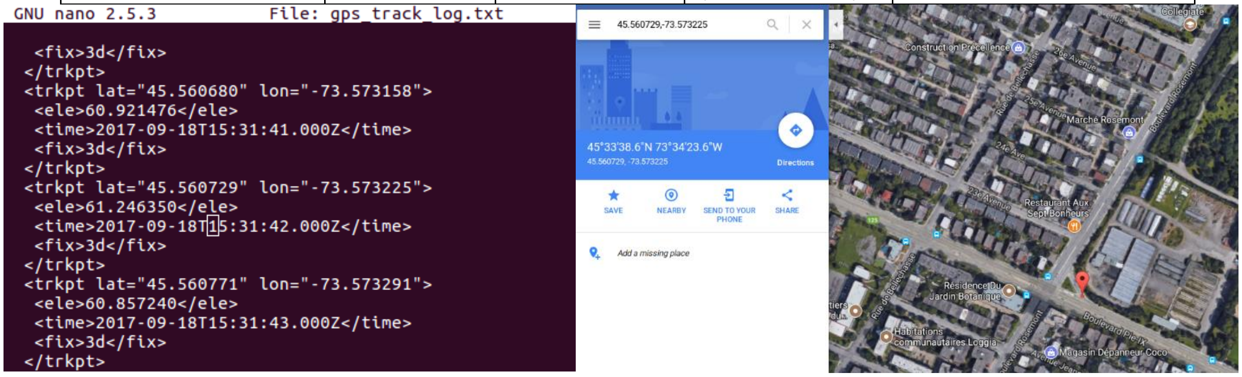
Further study still needs to be conducted to isolate device reliability.

4.1.3 Road surface and tree inspection vehicle field test

The road surface and tree inspection vehicle stores the videos from the cameras and GPS logs locally on the Mac Mini. Due to the huge amount of data generated, the data offload can be done manually or via high-speed WiFi when it is available. The field test results are presented in Table 20.

Table 20: Road surface and tree inspection vehicle field test storage results.

	Frame rate	Shutter time	Resolution	Storage consumption
Road surface 3D camera	2 fps	0.24ms	1024x768 (color)	9.6 Mbytes/s 35 GB/hr
Tree inspection 3D camera	2fps	0.24ms	960x600 (MONO 8 bit)	2.3 Mbytes/s 8.3 GB/hr



```
GNU nano 2.5.3              File: gps_track_log.txt

  <fix>3d</fix>
 </trkpt>
 <trkpt lat="45.560680" lon="-73.573158">
  <ele>60.921476</ele>
  <time>2017-09-18T15:31:41.000Z</time>
  <fix>3d</fix>
 </trkpt>
 <trkpt lat="45.560729" lon="-73.573225">
  <ele>61.246350</ele>
  <time>2017-09-18T15:31:42.000Z</time>
  <fix>3d</fix>
 </trkpt>
 <trkpt lat="45.560771" lon="-73.573291">
  <ele>60.857240</ele>
  <time>2017-09-18T15:31:43.000Z</time>
  <fix>3d</fix>
 </trkpt>
```

45.560729,-73.573225

45°33'38.6"N 73°34'23.6"W
45.560729, -73.573225

Figure 20: GPS logs and position of the truck.

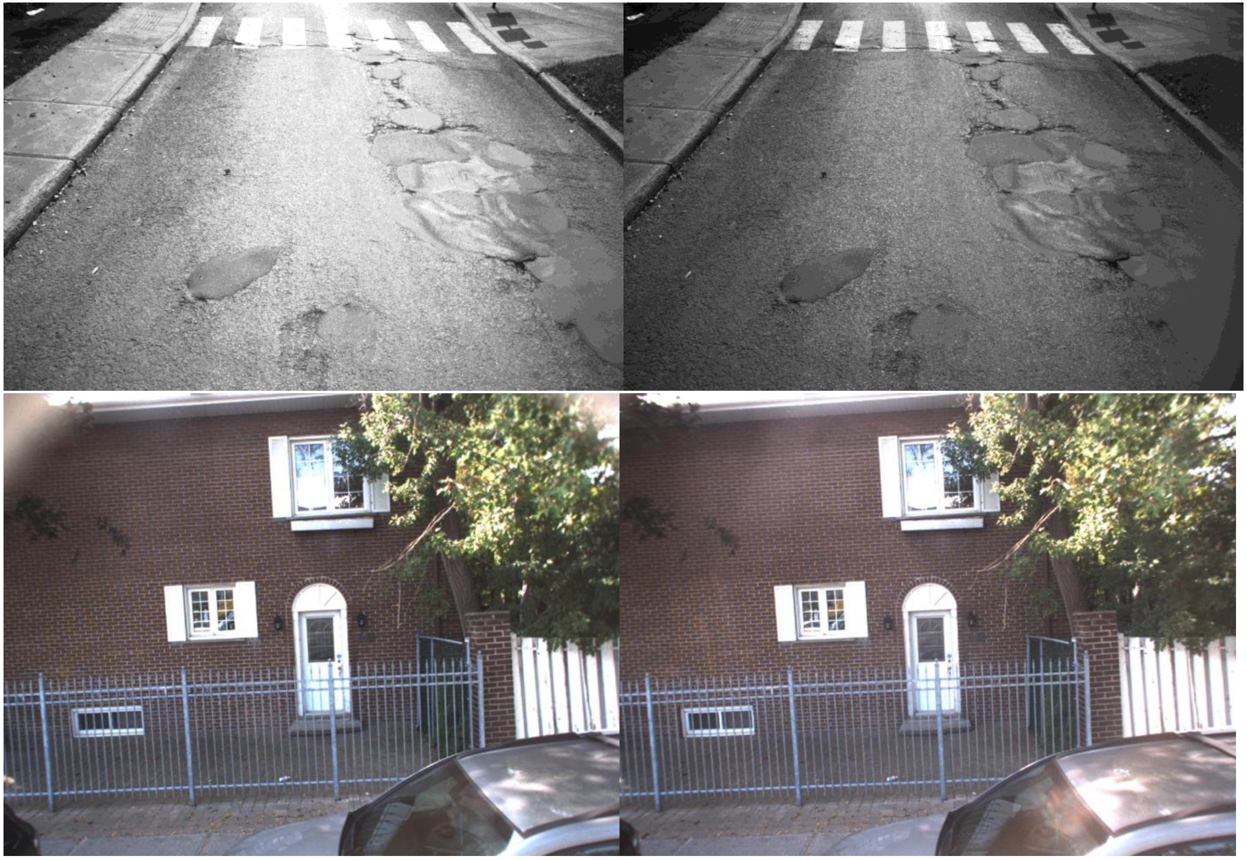


Figure 21: Footage from street inspection camera (top) and tree inspection camera (bottom) for both left and right channels.

It is noted that due to the power constraint, the current vehicle does not allow the installation of an UPS to back up the power for the Mac Mini for proper shutdown. Further developments can be done in the next steps for further optimize the operation of the system.
The position of the truck is continuously logged using the GPS sensor and is stored inside the Mac Mini installed inside the truck. The log file can be used to track the position of the truck latter as in Figure 20.
Figure 21 shows the footages from both the street and tree inspection cameras for both left and right channels. With the frame rate at 2fps, and vehicle travel speed at 40km/h (normal travel speed in urban area), a frame is taken approximately every 5.5m. For a standard configuration, the GPS logs location every seconds so the accuracy of the interested location can be in the vicinity of ±11m from the location obtained from the GPS log.

5. Conclusions

The results of the IoT smart city demonstration deployment highlight certain aspects that must be carefully considered during the development of a smart city structure and its feasibility. Special attention must be placed on deciding how devices are going to interconnect, what device capabilities are required in a specific deployment, and an awareness of deployment performance decrease.

Environmental and configuration factors were found to have strong effects on wireless data transmission rates. Wi-Fi devices, once configured to maintain stable connection in downtown Montreal, saw data throughput rates drop to below 20% of manufacturer advertised speeds in some cases. While the Wi-Fi radios did not experience a speed decrease with enabled encryption, not all devices may behave the same way. The ZigBee Pro radios pushed less than 50% of the max throughput when AES encryption was enabled vs disabled. When deploying a large-scale network structure, it is imperative that conservative estimations of device capabilities and guidelines relating to deployment be followed.
The deployed network proved to be relatively stable with devices maintaining connectivity for more than 95% of the time.

Further research into more communication technology is necessary for improved IoT smart city structure design and performance. Better directed antennas and communication mediums may decrease the amount of interference experienced in busy city sections and improve throughput possibilities to reduce bottlenecks and allow greater information transfer.

5 References

[1] A. Gupta and R. K. Jha, "Security threats of wireless networks: A survey," IEEE, Noida, India, 2015.

[2] Y. Zou, J. Zhu and X. Wang, "A Survey on Wireless Security: Technical Challenges, Recent Advances, and Future Trends," IEEE, 2016.

[3] C. M. Ramya, M. Shanmugaraj and R. Prabakaran, "Study on ZigBee technology," IEEE, Kanyakumari, India, 2011.

[4] L. Zhang, "A Retrospective View of Network Address Translation," IEEE, Los Angeles, 2008.

[5] R. Want, "An introduction to RFID technology," IEEE, 2006.

[6] K. H. Kim, E. Y. Choi and D. H. Lee, "Secure EPCglobal Class-1 Gen-2 RFID System Against Security and Privacy Problems," OTM, 2006.

[7] A. Juels, "RFID Security and Privacy: A Research Survey," IEEE, 2006.

[8] H. Niu, E. Taqieddin and S. Jagannathan, "EPC Gen2v2 RFID Standard Authentication and Ownership Management Protocol," IEEE, 2015.

[9] D. L. Mills, "Internet Time Synchronization: Network Time Protocol," IEEE, 1991.

[10] S. T. Watt, S. Achanta, H. Abubakari, E. Sagen, Z. Korkmaz and H. Ahmed, "Understanding and applying precision time protocol," IEEE, 2015.

[11] D. C. Mazur, R. A. Entzminger, J. A. Kay and P. A. Morell, "Time Synchronization Mechanisms for the Industrial Marketplace," IEEE, 2017.

[12] H. Guo and P. Crossley, "Design of a Time Synchronization System Based on GPS and IEEE 1588 for Transmission Substations," IEEE, 2017.

[13] D. G. Chandra, M. Kathing and D. P. Kumar, "A Comparative Study on IPv4 and IPv6," IEEE, 2013.

[14] M. Viswanathan, "Channel Capacity," in *Simulation of Digital Communication Systems Using Matlab*, Kindle, 2013.