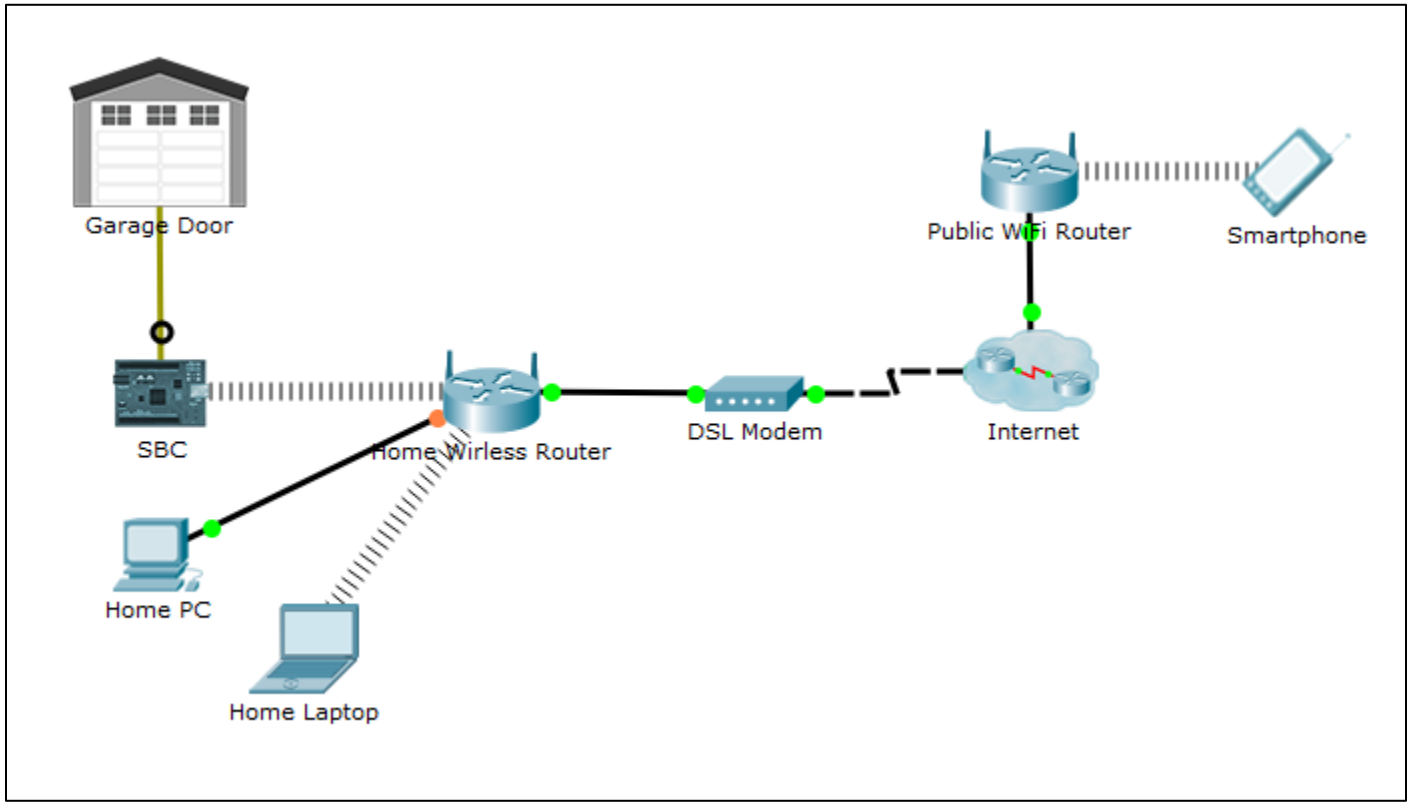
Packet Tracer - Case Study - Prototype & Test the Solution

Topology



Objectives

- **Explore Packet Tracer as a Prototyping Tool**

Background / Scenario

In this activity, you will have a chance to explore a Packet Tracer based IoT system prototype designed to allow a user to remotely check whether the garage door is open. You are encouraged to freely explore the prototype to gain a better understanding of the technologies used in the system.

Required Resources

- PC with Packet Tracer 7.1 or newer installed

Part 1: The Topology

Step 1: The Network

The network topology is designed to simulate a home which is connected to the Internet via a DSL-based Internet Service Provider (ISP). A wireless router provides wireless access to all the home devices (desktop computers, laptops, smartphones, etc).

The wireless router is also connected to the WAN via a DSL modem which in turn connects to the Internet. Note that the DSL modem is provided by the ISP.

The SBC device represents a Raspberry Pi computer and is connected to the **Garage Door**. Since the garage door has no intelligence or Internet connectivity of its own, the SBC is used to fill that gap. Similar to the other devices in the home, the SBC connects to the home wireless via the home wireless network.

The **Home PC** and **Home Laptop** were added to represent other home devices which are connected. Any of these devices can also be used to access the system.

The **Public WiFi Router** is added to represent free Internet connections through the city. Think of it as wireless hot spots present in libraries, coffee shops, workplaces, and more. The key here is that the user will also have a connection to the Internet.

The **Smartphone** represents the user when he or she is away from the home. In this scenario, Internet access would be achieved through that Smartphone. Notice that the user could also connect to the system from any other device, as long it was connected to the Internet.

Note: Python used in PT is an open source Python to JavaScript interpreter that is not updating to Python 3.0. For this reason there may be slight differences in the syntax between the code observed in PT and that in devices using Python 3.

a. Click the **SBC**.

b. Click the Programming Tab. On the left pane that opens, a file name is displayed, **main.py,** in this case. This is the program that will be running in the SBC on behalf of the **Garage Door**.

c. Packet Tracer displays the actual code stored in **main.py** on the right area of the window. This is an important window because you can edit, stop and start the program.

d. Click the **Run** button to execute the program. What happens?

_____

e. This code, written in Python, is responsible for periodically transmitting the status of the garage door to a server on the Internet. If the door is shut, the code sends out 0. If the door is open, 1 is sent to the Internet server.

Looking at the code, answer the following questions:

How often are door status updates sent to the Internet server?

_____

What is the IP address and port of the server?

_____

What are the variable names used to store the server IP address and server port number?

_____

What Python modules were imported into the code?

_____

Part 2: Using the System

To use the system, consider the following use case scenario: The user leaves the house and stops by a coffee stop on his or her way to work. During the stop, the user wonders if the garage door was shut or not. Because the system is already in place, the user can quickly access the Internet server address from his or her phone web browser to check the door status.

Step 1: Opening the Status Page From the Smartphone

a. Click the Smartphone and select the **Desktop** tab in the window that opens.

b. Navigate to **Desktop** > **Web Browser**.

c. In the URL address bar, type in **www.connthings.example**. This is the IP address of the Internet server used to collect the data sent by the SBC.

d. The page that opens in the **Smartphone** should display a green bar, communicating the garage door is shut.

e. With the status page still open on the Smartphone, open the garage door by holding down the ALT key and clicking the door. The door should open.

What is displayed in the Smartphone?

_____

The SBC which connected to the door updates the server which in turn, updates a local file with the status of the door. When a user connects to the server via a web browser, the server presents a web page built based on the door status stored in the disk.

Based on what you have learned in this course, what would be the role of the Internet server used in this Packet Tracer Prototype?

_____

f. The system is designed to allow any device that is connected to the Internet to use it. Try opening the door status page from the Home PC and from the Home Laptop.

Does it work?

_____

What is the benefit of accessing the system from within the house?

_____

Part 3: Challenge Questions

a. Expand the Internet Cloud and explore its contents. Try locating the DNS server and the web server that receive the updates from the SBC.

b. Locate the Multilayer Switch. What is its function in this prototype?

_____

c. Add more devices to the Home Network and try to access the system from them.

d. The system currently doesn't store garage door status but it could be a good future feature. Can you think of one benefit of storing the door status over time?

_____

e. Can you describe, as an overview, what changes would have to happen for the current system to track the status?

_____

f. What crucial IoT feature is missing in this prototype?

Part 4: Reflection

The system prototyped here is simply a proof of concept but can be used to present and even refine some aspects of a project. By nature, Packet Tracer prototypes are very inexpensive as it allows for testing projects that require complex network topologies. Also, Packet Tracer's Python support makes it easy to improve and eventually port any code to the actual devices when the testing phase is complete.