INFOSNIFFER

Host to IP, IP to Host, Port and Protocol to Service
Name, Service Name and Protocol to Port

Abstract
This tool uses Winsock 2 API calls which get information about an IP/Host, Services and Ports.

Justin Cervantes
Cervantes.jfa@gmail.com

Contents

About InfoSniffer

This InfoSniffer application is a simple GUI with 4 main functions being utilized from Winsock 2: gethostbyname, gethostbyaddr, getservbyname, getservbyport. They are wrapped in functions which are triggered by the on_pushButton_clicked() method. The wrappers functions are: hostNameToIP, ipToHostName, serviceNameProtocolToPortNumber and PortNumberToServiceNameProtocol, respectively. As the API's and wrappers names' imply, the application will allow you to convert a host name to IP, an IP to a host name, a service name to a port number, and a port number to a service name.

The application was built in Qt Creator for Windows as a desktop application.

Diagram
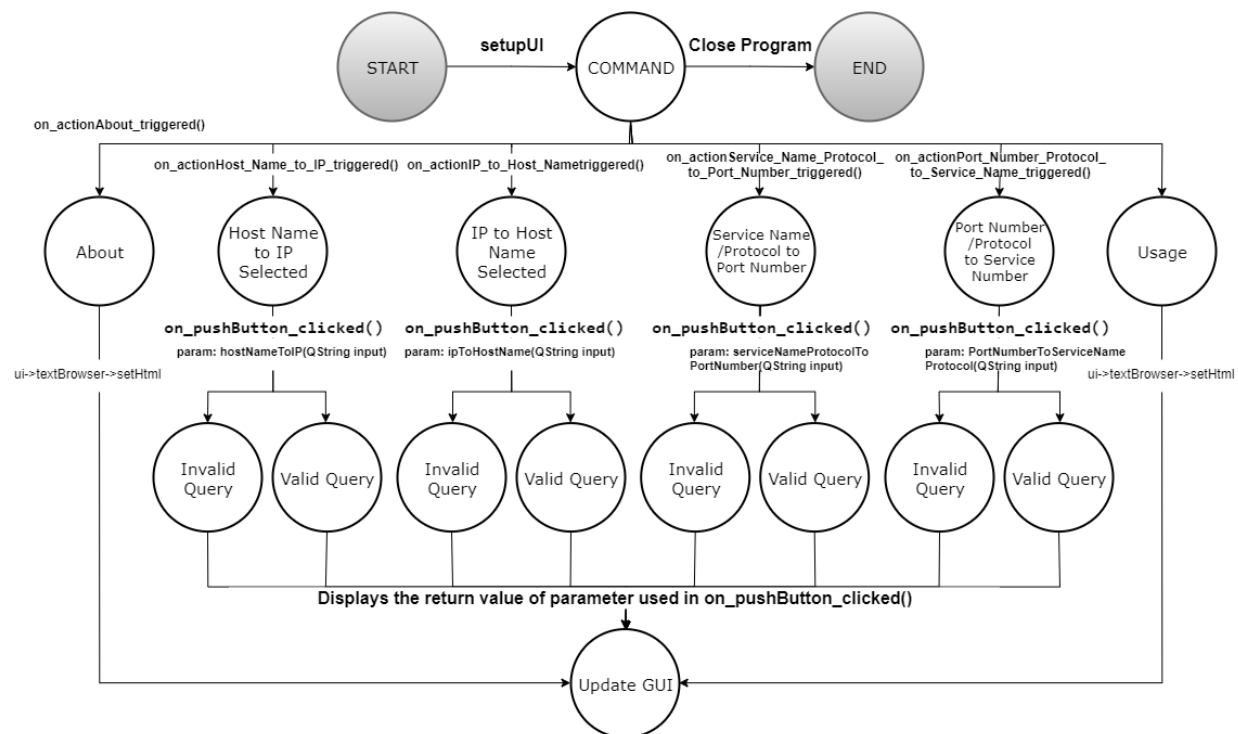
Information

This state diagram shows a general flow of how the UI works. When the program starts, the user enters command mode, where they can interface with the menu items, an input box, and a query button. This state diagram only details the main use case, wherein the user selects an action from the menu first, fills in input into the input box, and then pushes Query.

Other scenarios, which are not outlined in this state diagram for brevity, include the user pressing the Query button before making a selection (this action updates the GUI with an error message), the user pressing the Query button without anything in the input field (this action updates the GUI with an error message), and the user making an alternate selection or action after making a first selection (ultimately, the user is always in the command mode state).

Pseudocode

Application Class (application.cpp)

on_actionHost_Name_to_IP_triggered

Set the menu action item to checked, and uncheck the other 3 menu action items to ensure that only one item (the selection) can be checked at a time.

on_actionIP_to_Host_Name_triggered

Set the menu action item to checked, and uncheck the other 3 menu action items to ensure that only one item (the selection) can be checked at a time.

on_actionService_Name_Protocol_to_Port_Number_triggered

Set the menu action item to checked, and uncheck the other 3 menu action items to ensure that only one item (the selection) can be checked at a time.

on_actionPort_Number_to_Service_Name_Protocol_triggered

Set the menu action item to checked, and uncheck the other 3 menu action items to ensure that only one item (the selection) can be checked at a time.

on_actionAbout_triggered

Update the UI's text browser with information about the application.

on_pushButton_clicked

Check if something is in the input box and if nothing's there, update the UI's text browser with an error message.
Check what action is currently selected.
Depending on the current selection, run the appropriate wrapper function with the input field's current text as the parameter.
If nothing is selected, update the UI with an error message.

getCurrentAction

Check each of the UI's menu action items one by one... when you find a menu item that is checked, return the name of the menu item.
If nothing is selected, return an empty string.

on_actionShow_Usage_triggered

Update the UI's text browser with information about the application's usage.

API Wrappers (session.cpp)

hostNameToIP
Initiate use of the winsock dll by a process.
Query the hostname and return an error message if the API fails.
Return a string with each known IP address.

ipToHostName
Initiate use of the winsock dll by a process.
Query the IP and return an error message if the API fails or an improper input was received.
Return a string with the hostname.

serviceNameProtocolToPortNumber
Initiate use of the winsock dll by a process.
Get the service by name and return an error message if the API fails.
Return the port number.

PortNumberToServiceNameProtocol
Initiate use of the winsock dll by a process.
Get the port by number, and return an error message if the API fails.
Return the service name.

File Structure

The application will have one driver which will be the entry point into the program called main.cpp. The application's gui and related code will be in the application.cpp/application.h files, and the API's wrappers will be contained in a session.cpp/session.h file.

As the program is being built using Qt Creator, the file will also contain any .ui files and .pro files for the project.

Starting the Program

To start the program, double click the InfoSniffer executable on a Windows machine.