Offensive Network Security

Assignment 4
CIS 4930/5930
100 points
Due: 02 May 2014

1. Talk to the executable `byteorder.x` to obtain the following answers. A client file has been provided to help talk to the provided server (`byteorder-client.py`).
    a. What port is the service listening? How did you determine the listening port?
    b. Implement the function `send_answer()`.
    c. List all the questions your client had to answer and the answers your application provided for each question (this can be the final list after implementing the function). Please give the integer given and your provided answer (in hex format).
    d. The client and server are using two different message structures to send and receive data. What are their fields and how are they used? Is there any way to reduce the data sent in messages? (HINT: You may modify the python client to print data)
    e. What is the final secret value?
    f. Send erroneous data to the server to cause it to crash. What values caused the server to crash? If none of the data you sent caused the application to crash still list all the erroneous data sent to the application.

2. A client executable, `ipkiss.x`, has been obtained from a network in which it was supposedly transferring data out of the network. During the duration of this client's existence on the network no data captures were obtained. It appears the server for this client is now down. Answer the following questions by auditing `ipkiss.x`.
    a. Run the program `strings` on the binary and list the results.
    b. What was the domain name this client was trying to connect? How did you determine the domain name? Did it appear in the list of strings or did you have to use other means to determine domain name? Any ideas why the domain name did not appear using the `strings` utility?
    c. How did you initiate a connection with the client?
    d. What is the initial message from the client to the server?
    e. What is the server's answer to the client's initial message?
    f. What information was provided after the initial setup?
    g. Send erroneous data to the client to cause it to crash. What values caused the client to crash? If none of the data you sent caused the application to crash still list all the erroneous data sent to the application.