Michael Colistro

0488092

COMP-4473

Bonus 2

Dr Sabah Mohammed

This is the regular expression that I ended up coming up with:

/^(?:(?:25[0-5]|2[0-4][0-9]|[1]?[0-9]?[0-9])\.){3}(?:25[0-5]|2[0-4][0-9]|[1]?[0-9]?[0-9])$/

The test strings that used to test the regex are here:

1) 172.16.254.1

2) 255.255.255.255

3) 315.255.255.255

4) 172.0.0.1

So with the regex in the first parenthesis , /^(?:(?:25[0-5]|2[0-4][0-9]|[1]?[0-9]?[0-9])\.), what its doing is first checking to see if the byte starts off with 25 and if it does it makes sure the third digit is within the available range of 0-5. If the byte doesn’t start off with 25 it checks if it starts with 2 and then gives the available range for the next 2 digits. Lastly if those two previous conditions were not met it checks the first digit to see if it is a 1, I put a ? after the 1 so that it satisfies the case of if there isn’t one there. So when that condition is met it then checks the next digit to make sure its in between 0 and 9 or if it is even there with another ? and then we check the last digit to make sure it is also between 0 and 9. The last square bracket does not have a ? after it because it has to have at least one digit. After all said is done it then checks for the (.) period. Next is the {3} which at the point rechecks each byte in the IP 3 times until it gets to the last byte and this is called: (?:25[0-5]|2[0-4][0-9]|[1]?[0-9]?[0-9]). This does the same thing as the first parenthesis except check for a period at the end. Those test strings are the ones I used to make sure the regex was working correctly, all worked except #3 which was intended.