Colin Keenan

3-11-2020

ECE 5200 Exercise 2

Total Points: 50 points

## Analysis Questions of modulo

1. (5%) Compute all powers of 3 (3, 32, 33, …) modulo 17. Note that 33 = 27 must be represented as an integer between 0 and 16 or 27 = 10 (mod 17). What is the smallest integer n such that 3n = 1 (mod 17)? What is the smallest k such that 3k = -1 (16) modulo 17? Do all the powers of 3 modulo 17 show all the integers 1, 2, .., through 16 in some different order?

Powers of 3 (mod 17):

[3, 9, 10, 13, 5, 15, 11, 16, 14, 8, 7, 4, 12, 2, 6, 1]

Smallest integer n such that 3^n = 1 (mod 17):

3^16 (mod 17) = 1

Smallest integer k such that 3^k = -1(16) (mod 17):

3^8 (mod 17) = 16

All powers of 3 (mod 17) sorted:

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16]

1. (5%) Compute all powers of 3 modulo 26. Do all the powers of 3 modulo 26 show all the integers 1 , 2, …, 25 in some order? Now consider all powers of 2 modulo 26. Is any power of that equal to 1 modulo 26? Can you explain why?

Powers of 3 (mod 26):

[3, 9, 1]

Powers of 2 (mod 26):

[2, 4, 8, 16, 6, 12, 24, 22, 18, 10, 20, 14]

Smallest integer n such that 2^n = 1 (mod 26):

Not Found with 1000001 iterations

The reason 2^n (mod 26) cannot equal 1 for any value of n because 2^n with any value of n is an even number, so when you modulo 26 an even number, it cannot be odd.

1. (15%) In page 117 of Stalling’s 6th edition, table 4.7(b) shows how two polynomials in GF (23) multiply modulo x3 + x + 1. In particular, it shows that when x2 multiplies x2, the result is x2+ x.

**Redo** table 4.7(b) using a different irreducible polynomial x3 + x2 + 1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | 0 | 1 | x | x+1 | x2 | x2 + 1 | x2 + x | x2 + x + 1 |
|  | x | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 0 | 000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 001 | 0 | 1 | x | x+1 | x2 | x2 + 1 | x2 + x | x2 + x + 1 |
| x | 010 | 0 | x | x2 | x2 + x | x2 + 1 | x2 + x + 1 | 1 | x+1 |
| x+1 | 011 | 0 | x+1 | x2 + x | x2 + 1 | 1 | x | x2 + x + 1 | x2 |
| x2 | 100 | 0 | x2 | x2 + 1 | 1 | x2 + x + 1 | x+1 | x | x2 + x |
| x2 + 1 | 101 | 0 | x2 + 1 | x2 + x + 1 | x | x+1 | x2 + x | x2 | 1 |
| x2 + x | 110 | 0 | x2 + x | 1 | x2 + x + 1 | x | x2 | x+1 | x2 + 1 |
| x2 + x + 1 | 111 | 0 | x2 + x + 1 | x+1 | x2 | x2 + x | 1 | x2 + 1 | x |

1. (15%) The textbook shows you GF (p) and GF (2n).

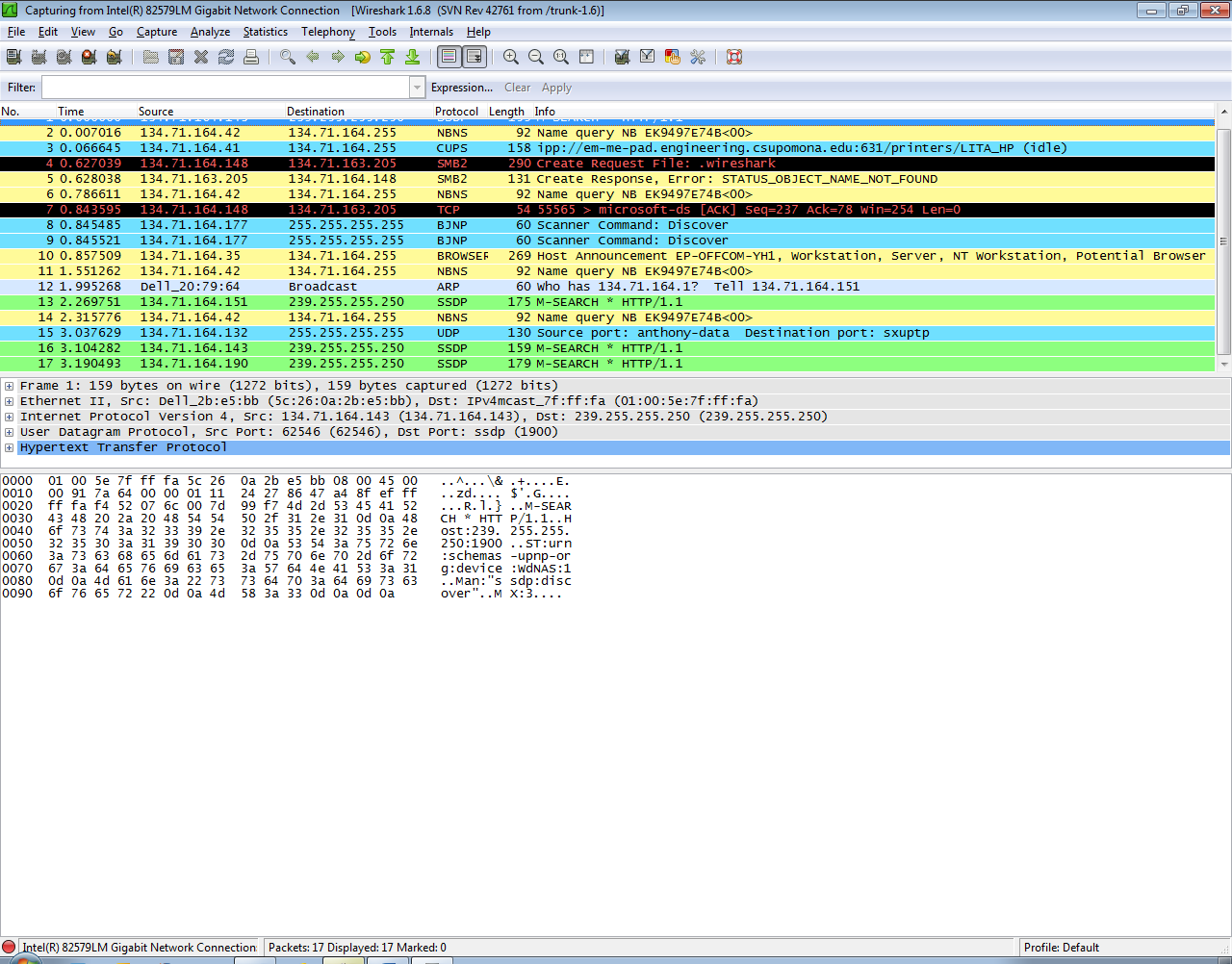
Now let’s do GF(32).

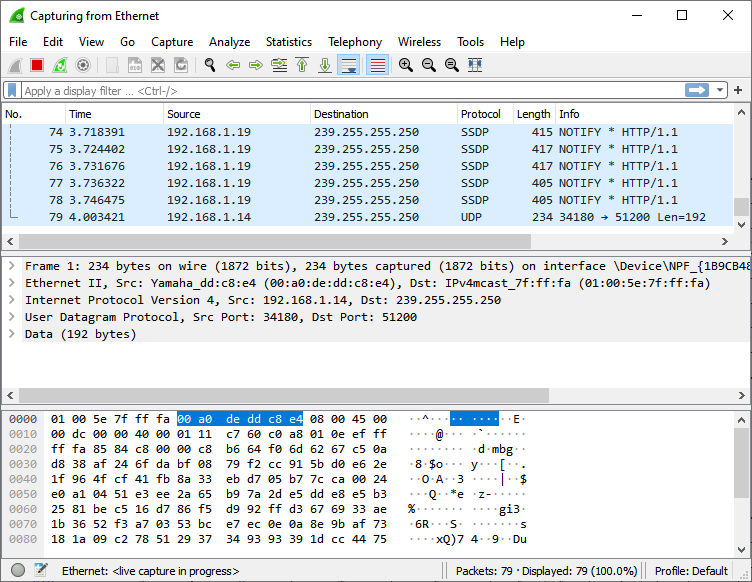
We need an irreducible polynomial f(x) over GF(3).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | Polynomial | Reducible/Irreducible | 1. What is the minimum degree of such polynomial? Degree 2 2. Find such a polynomial f(x): See table to the left 3. With the solution you have from (b), generate the multiplication table similar to table 4.7 (there are 9 entries in GF(32) in contrast with GF (23) which has 8 elements.   I am still very confused on how to tell the reducibility or irreducibility of a polynomial and as such I cannot generate this table. |
| 0 | 1 | 0 | x | Reducible (0) |
| 0 | 1 | 1 | x+1 | Reducible (2) |
| 0 | 1 | 2 | x+2 | Reducible (1) |
| 0 | 2 | 0 | 2x | Reducible (0) |
| 0 | 2 | 1 | 2x+1 | Reducible (1) |
| 0 | 2 | 2 | 2x+2 | Reducible (2) |
| 1 | 0 | 0 |  | Reducible (0) |
| 1 | 0 | 1 | +1 | Irreducible |
| 1 | 0 | 2 | +2 |  |
| 1 | 1 | 0 | +x |  |
| 1 | 1 | 1 | +x+1 |  |
| 1 | 1 | 2 | +x+2 |  |
| 1 | 2 | 0 | +2x |  |
| 1 | 2 | 1 | +2x+1 |  |
| 1 | 2 | 2 | +2x+2 |  |
| 2 | 0 | 0 |  |  |
| 2 | 0 | 1 |  |  |
| 2 | 0 | 2 | 2+2 |  |
| 2 | 1 | 0 | 2+x |  |
| 2 | 1 | 1 | 2+x+1 |  |
| 2 | 1 | 2 | 2+x+2 |  |
| 2 | 2 | 0 | 2+2x |  |
| 2 | 2 | 1 | 2+2x+1 |  |
| 2 | 2 | 2 | 2+2x+2 |  |

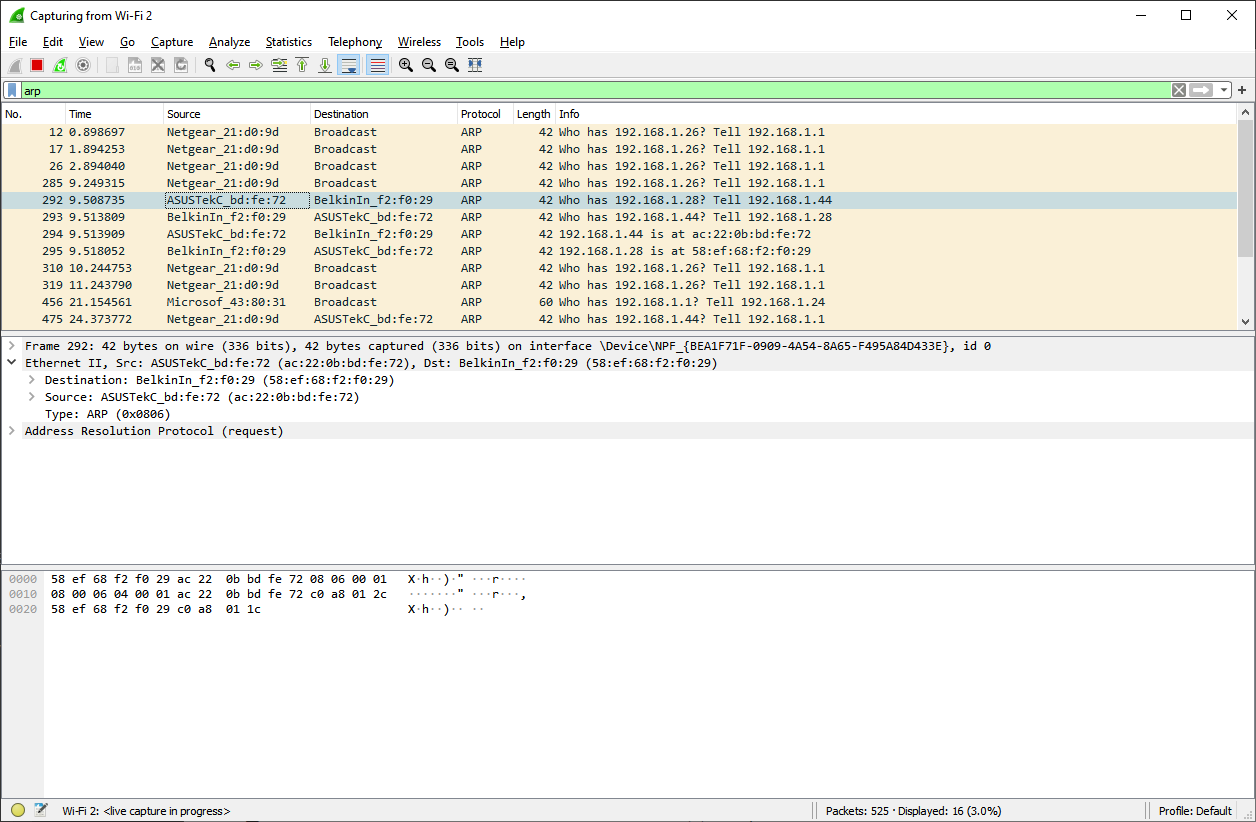
Tool and running code

1. (10%) Download Wireshark from <http://ww.wireshark.com> and run that.
2. Start capturing data like below:





1. Do some network activity and see what wireshark captures. Activity could be starting a command window and type ping [www.csupomona.edu](http://www.csupomona.edu); could be starting an internet browser etc. Try to find / identify what kind of frames correspond to your network activity.



After arp -a command sent to command prompt. As you can see my computer is polling the entire networking asking for all of the IP and MAC addresses on the network. This is a lot in my case because of the amount of IoT devices on my network.**Extra Credits (43%)**

**Modulo:**

~~E1. (8%) Analysis: Generalize Q6 and Q7 to powers of a modulo n for any positive integer a and positive integer n (a and n may not be prime numbers or relatively prime). Explain when will the power of a be first equal to 1 modulo n (or maybe that never happens for certain combinations of a and n).~~

E2. (10%) Programming: Write a (computer) program that inputs a and n and computes and displays powers of a modulo n.

**def** powMod**(**a**,** n**):**

i **=** 1

ans **=** **[]**

while(True):

temp **=** pow**(**a**,**i**,**n**)**

**if** temp **not** **in** ans**:**

ans**.**append**(**temp**)**

**if(**i **==** n**):**

**break**

**else:**

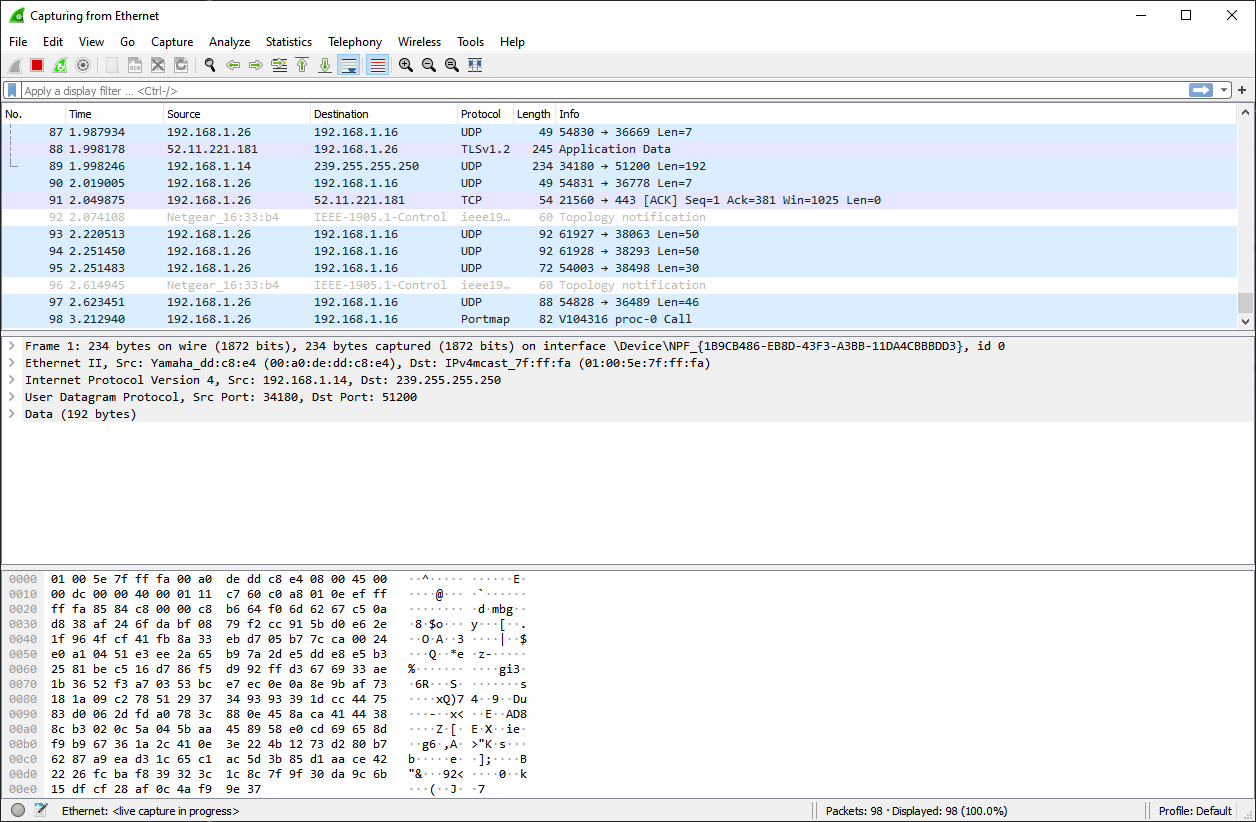
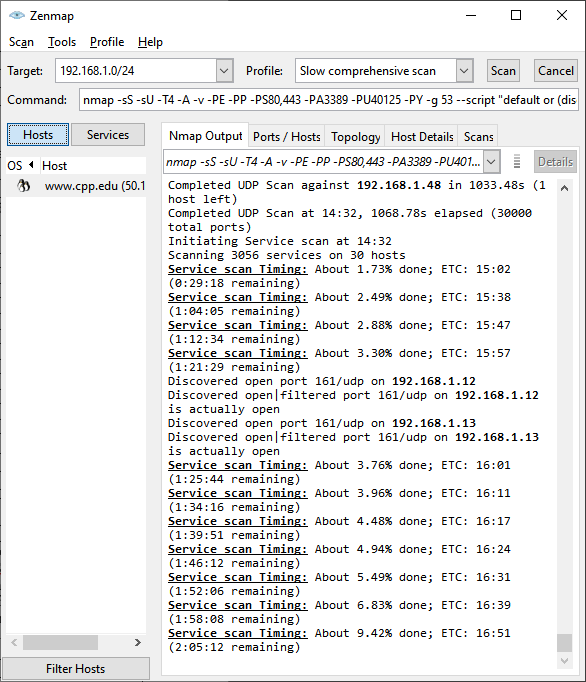
i **+=** 1

**return** ans

**Security:**

E3. (15%) Study network sniffer / scanner tools such as Nessus, Nmap and write on what you find (on your computer / network, not what you see on internet, books etc.)

Nessus appears to be a closed source network sniffer/scanner/monitor/manager meant for enterprise usage versus the consumer usage and open source nature of Nmap and WireShark. As a result, I was unable to download or try Nessus, but Nmap was successfully installed and compared to Wireshark. Wireshark actually is built using Nmap in the backend and is just an alternative GUI. The GUI provided with the Nmap download however is Zenmap, which has a few features different from Wireshark, such as being able to look at certain targets and scan locally over multiple ports and multiple addresses, as well as over the internet. This differs from Wireshark in that Wireshark focuses on what packets are being received and sent from the perspective of the host computer’s NIC.



E4 (10%) Study how Wireshark may be used in Intrusion Detection / Prevention etc.

Wireshark can primarily be used as a source of intrusion detection though the use of its filters. Wireshark has two types of filters, one used in the capture of packets, and one used in the display of packets. The filter used in the capture of packets helps to minimize the amount of clutter in the display by filtering out only a select protocol, but in the case of intrusion detection, where the intruder may be trying over multiple protocols, this filter is not particularly useful. The second filter however, the filter to display the packets captured is, as they can be configured to detect patterns such as successive requests in the case of a DDOS, or the detection of errors. Unlike IDS/IPS though, Wireshark does not have automatic alarms for these types of patterns and need to be analyzed manually or with other software. Wireshark however does have some useful tools built in such as the IO graphing utility to show a data rate/time plot of various connection/protocols/etc. This IO graphing tool allows visualization of the data being sent and received on the computer and would also visualize any oddities or discrepancies compared to a normal data flow.

Source: <https://pdfs.semanticscholar.org/38f0/641fc38868aad84a9008b13769afbc31c3b1.pdf>