CMSC 417 Computer Networks **Fall 2015**

First Third-Term Exam

Closed book and notes; In class *Thursday, Oct. 8rd*

⊕ *Do not forget to write your name on the first page. Initial each subsequent page.*
⊕ *Be* **neat** *and* **precise**. **I will not grade answers I cannot read.**
⊕ *You should draw simple figures if you think it will make your answers clearer.*
⊕ *Good luck and remember, brevity is the soul of wit*

- All problems are mandatory
- I cannot stress this point enough: **Be precise**. If you have written something incorrect along with the correct answer, you should **not** expect to get all the points. I will grade based upon what you **wrote**, not what you **meant**.
- Maximum possible points: 50 + bonus.

Name: _____

Problem	Points
1	
2	
3	
4	
5	
Total	

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2	4	8	16	32	64	128	256	512	1024	2048	4096	8192	16384	32768	65536

1. Nomenclature

   (a) Describe the following terms: (2 points each)

   - Certificate Authority

   - **I**nternet

   - Poisoned Reverse

   - Default-free router

   - BGP Weight Attribute

2. Routing

(a) (How) Does the space requirement of Distance Vector routing depend on triggered vs. periodic updates?(2 points)

(b) What are the uses for sequence numbers in link state updates? (3 points)

(c) Let N be the node set. At node s, $C(t)$ is the best-known cost t, and $l(t)$ is the link cost to node t (∞ if t is not a direct neighbor). Dijkstra's algorithm at node s proceeds as follows:

```
initially
    M ←s ; ∀ n ∈ N-M, C(n) ←l(n)

while M ≠ N
    M ←M ∪ w such that C(w) is minimum ∀ w ∈ N-M
    /* recompute */
```

Using the notation above, describe the *recompute* step. How does its complexity vary with different data structures? (3+2 points)

3. Internet Protocol

(a) Consider the following excerpt from host `ringding.cs.umd.edu`:

```
ringding% /sbin/ifconfig -a
eth0      Link encap:Ethernet  HWaddr 60:EB:69:82:49:D3
          inet addr:128.8.129.2  Bcast: [deleted]  Mask:255.255.254.0
          ...
```

What subnet is `ringding` on? What is the broadcast address? What is a valid address for `ringding`'s first hop router? (3 points)

(b) Suppose you were to split `ringding`'s current subnet into four. What would be the mask and the subnet numbers? (2 points)

Mask	Subnet 0	Subnet 1	Subnet 2	Subnet 3

(c) Suppose you need fragment a IP datagram, without header options, with 1280 payload bytes to be transmitted over a link with 580 byte IP MTU (excluding link layer headers). Fill in the values below assuming maximum sized fragments. (Max. 2 points) (Each unique incorrect value will lose $\frac{1}{2}$ point.)

Identification	Offset	MF	DF	Total len.
417				

(d) IP reassembly code receives a datagram with previously unseen Identification=651, Total Len 1024 bytes, MF flag=0, and offset=8191. What should it do? Why? (2 points)

(e) What if the MF bit in the previous datagram was set to 1? (1 point)

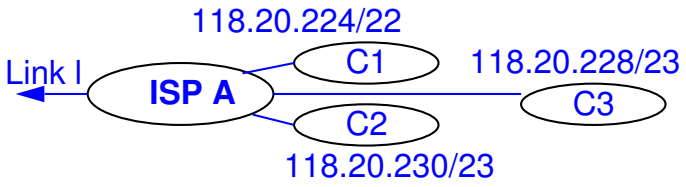
4. CIDR, BGP

(a) What is the difference between a *transit* and *multi-homed AS*? (2 points)

(b) What is a *BGP Speaker*? How many does a AS require? (1+1 points)

(c) Suppose the BGP for AS 11 learns of three different routes to prefix 31.13.31/16 (AS 417). How many of these routes *must* AS 11 advertise to its peers? (2 points)

(d) What prefix should ISP A advertise on link l to the Internet? C_i are customers of ISP A with the address allocations as shown. Note: 224 = 128+64+32. (1 point)



(e) What should ISP A advertise on link l if the address allocation for C1 changes from 118.2**0**.224/22 to 118.2**1**.224/23?(3 points)

Certificates, Implementation

5. (a) What are self-signed certificates? Why are they considered inadequate security for secure web transactions? (3 points)

   (b) Give two positive uses for self-signed certificates. (2 points)

   (c) `typedef int(*callback)(void); typedef unsigned int uint;`
   `uint nbRead(int socket, char *buf, uint len, uint yeildTime, callback cb);`

   The non-blocking read function `nbRead` prototyped is passed a connected TCP socket `socket`, an adequately sized buffer `buf` to read into, and a number of bytes to read `len`. `nbRead` returns when `len` bytes are read, and calls the callback function `cb` every `yeildTime` milliseconds while waiting for data. `nbRead` returns the number of bytes read, with (possibly partial) data in `buf`, if any invocation of `cb` returns 0.

   Provide (pseudo-)code for `nbRead`. If you use non-POSIX functions, explain what your assumptions are about these functions. You will be graded on both the simplicity and efficiency of your solution. (5 points)