Networking Final Project
Daniel Graham PhD

Name: _____
Name: _____
Name: _____
Name: _____

This project description contains 3 pages (including this cover page) and 5 questions. Total of points is 100.

This is projects has been designed to be fun and educational. OK, so without further ado, here is the problem: You are contracted by a secret government agency to setup a black-site. This black-site must support some basic functionality of the internet but it must not be connected to internet. (So that it is harder to hack, you know.) The Black-Site will only allow outgoing connections which must pass through an encrypted FM broadcast station. This secret agency does not trust anyone. So all the software will need to be developed from scratch. This includes, web servers, DNS Servers and the FM broadcasting station. Don't worry thought they will provide you with switches that implement the DHCP server. (AKA. Just use UVA network or your own homenetwork. Just connect to any access point on campus and the campus DHCP servers will give you an IP address.) With the exception of the SDR radio receiver all of your code will be running at the application layer.

Great so that is the big idea. The figure below shows the proposed topology of the network and the remaining questions in the document are designed to help you get there. (Remember use UVA's network to provide the physical infrastructure.)

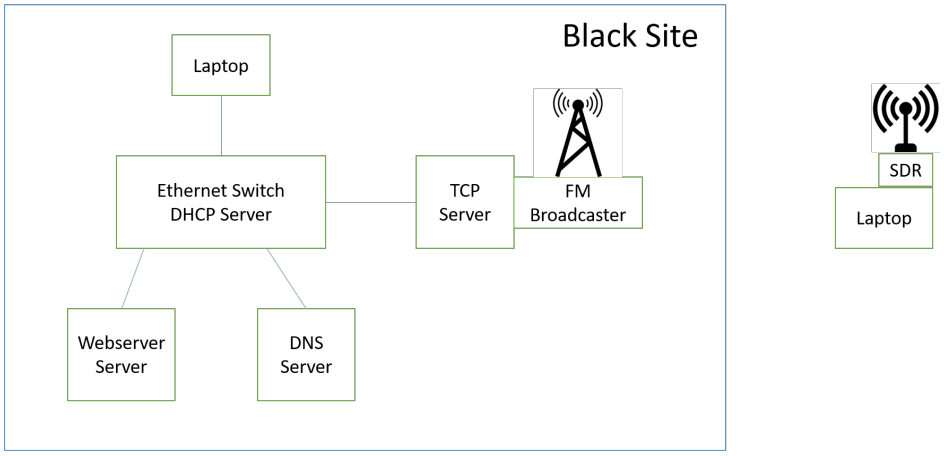

Figure 1: Black Site Topology

Grading Table

Question	Points	Score
1	15	
2	20	
3	20	
4	20	
5	25	
Total:	100	

1. (15 points) Implement a base web-server that responds to all web request with a simple HTML page that says:"Secret black Site" and displays the IP-Address of the requester.

2. (20 points) The users of the back-site don't know the ip-address of all the web-servers at the site. Instead they want to simply remember the names of the servers at the site. Implement a DNS server that provides the IP-Address for the web-server you implemented above. (The URL for the server should be blacksite.secret). Test your DNS server by connecting it to your network and typing blacksite.secrete into your browser. (You can configure the machine running your web-server to have a static ip-address. (This way you can simply hard code the address of the DNS server)

   Great people can now connect to your network and interact with servers on your network using their browser. Now let's allow your black site to send data to another external site using a custom FM broadcast station.

3. (20 points) The network contains a machine that is connected to the FM broadcaster. Any information that is played on the soundcard of this machine will get broadcasted on FM 87.7. Write a TCP server that accepts incoming request on port 4000 and sends the Bytes directly to the sound card on the machine. Use your software defined radio and SDR Sharp to verify the bytes are getting transmitted correctly. (You will should implement your own TCP Client that communicates with the TCP server)

4. (20 points) You want your receiver to be able to decode information that you send so you decide to create your own packet structure. You packet should contain the header length, check-sum and payload size.

5. (25 points) You don't want other people to be able to decode your packet so you encrypt the payload using a onetime pad. You can assume that you will generate a key.txt file that you can share between the machines which is the length of message you generate yourself. (We will send the packets 30 times just in case they got corrupted with a 10 second pause in between.)

   Great Awesome you have done it.

1 Bonus (50 points and an A)

Now for the bonus points. Our FM station is broadcasting our encrypted packets but we want to do more. We want to be able to also decode these packets. Implement the python program that will interface with your SDR radio and demodulate the signal, decode packets and decrypt the payload.