Idea: Use CORE and create an insecure network

- Create a proper CORE network with configured protocols and some security measures
    - Then export the CORE network and run on python
    - A simple network between <10 computers with a server, and have the proper protocols and security measures within hubs, switches, routers, etc.
- Find the insecurities within the python script (need to write ourselves)
        - Insecurities: Choose 3 routes to go
            - Faulty Certificates (with an insecure CA/ports)
            - Mike in the Middle attack → can change key or capture data
            - Unpatched software
            - Poor encryption
        - Learn how to hack from exploits from different resources
        - Find a way ourselves to hack from the exploits in the emulated network
    - Find ways to hack the network through the insecurities for the scenario(identify theft or hack into bank system)
    - Try to create something within one of the emulated servers that has some fake domain to hack
- See if running the network on Python gives the CORE visual, otherwise create some sort of visual
- Find ways to make this work in python
- Create an interface for a player to hack the emulated network
- Create a server for the game (xammp server on the “server” computer)
    - Create explanations for how to install
        - (Try to make it easy)
- Create explanations, hints, objectives and demonstrations within the game

- Last thing: name the game

- If time: create a decent looking UI
    - Anticipate what user is doing

Anuj Gajjar, Ananya Gopalan, and Madhu Sekar

Draper Laboratory

July 31, 2017

A Prototype of an Educational Network Security Game

Creating a computer game that is both informative and enjoyable requires a vision of both the purpose and the intended audience. As high school students ourselves, we can completely relate to the frustration and difficulties that may come with trying to learn something new only through lectures or reading material. In order to strike the balance between both having fun and gaining knowledge, in our case about Network Security, we tried out multiple platforms and decided to integrate the features we found to be most helpful and enjoyable out of all of them into one entity.

After numerous deliberations, we set our sights on utilizing the Core Network developed by the Navy as our base. We decided on creating a simple network within the program itself, including a few computers, along with a server and typical protocols that would be found in a common network. Following the creation of this network, we will export the code in python and depending on how the code runs through python, we will then have a more defined approach for creating the game. While working with the network, we will create numerous insecurities in the network, ranging from faulty certificates, unpatched software, and poor encryption to act as targets for the “hacker”. After creating these insecurities, we will then proceed to the creation of the game.

As the purpose of the game is to hack into a sensitive virtual network, taking advantage of the server’s vulnerabilities would be the only method to do so. From the python script, which

is to be modified ourselves, the hacker has to locate insecurities in order to break into the network. The user will be given the option of finding and choosing at least five different insecurities. Properly exploiting these weaknesses will enable them to hack into the virtual network. A variety of insecurities that will be presented to the hacker include: faulty certificates (through breached CA's), insecure ports, unpatched software, and poor encryption. The player will also be able to perform a man in the middle attack if they choose to do so. Additionally, a fake domain is also going to be created within one of the emulated servers as another platform of the network that can be taken advantage of.

For the creation of the actual game, we will first see if running the exported python script of the emulated network will run through the CORE simulation or runs from python directly. After that is determined we will create the hacking element of the game, and basic user interfaces and visuals. If the script uses the application, we will have to find a way to make the game run through python. If the script runs through python, then we will directly proceed to creating the game. While creating the hacking portion of the game, we will also learn of the ways a hacker can get into a network through its insecurities. Using that knowledge we will create a proper interface for a user to hack the network itself. The interface is going to consist of a way to look at the network, various ways to find the insecurities of the network, and will provide the user various techniques to breach the network. Once all of this has been completed, we will have successfully created a prototype of a network security game. The next steps includes the creation of a server for the game to run on, which will likely be a Xammp server, and a creation of a method for the game to be relatively easy to install from Github. If there is time remaining, we will also try to improve the game in various ways which include the creation of better visuals, an improved UI that anticipates the actions of a user, and finally include explanations, hints, and descriptions of networking and cryptography concepts within the game.