Amer Abdelaziz , Jay Patel and Al Mundhar Al Hadhrami

November 28, 2018

CNT 4004.001 Computer Networks I

Dr. Ken Christensen

Networks I Final Project

Hidden Web Server Using Port Knocking

By: Amer Abdelaziz , Jay Patel and

Al Mundhar Al Hadhrami

Table of Contents

1.0 : Introduction

Throughout the history of Computer Networking, many protocols have been defined and implemented successfully. Some of these protocols are used everyday such as the OSI protocol and the TCP protocol, both have been used in the development of the internet and have been used to improve network connectivity. Port knocking is a protocol that has been used to enhance the security of network infrastructure. Port knocking is used to ensure that only certain ports are open upon request. A firewall is similar to port knocking in the sense that firewalls acts like an ip packet filter and ensures that only a handful of packets get across the firewall to make a connection. The configuration of the firewall relies on the source ip address as there are exists ip tables to determine whether an packet with a source ip address isn't in the ip table of the firewall.

1.1 : Basics of Port Knocking

Communicating two devices within the same network requires that either the source or the destination includes an ip address and a port number in the packet that is being sent from the source to the destination. An ip address is analogous to a street address and the port number is analogous to "care of" field of the shipping label of the shipped good. The IP address determines where the packet goes to as every registered device in a network has an ip address and the port number determines which application does the data go to in a "station". A Port is used to determine where the data must go in the machine in terms of application. Every application has a port number associated with it, the range of ports are between 0 and 65535. For example, the port number for SMTP or better coined to be known as "Simple Mail Transfer Protocol" uses port number 25, HTTP uses port 80 for web content delivery, and Secure Shell Hosting uses port

number 22. When theses programs are using these ports, it imputes that these programs or applications are open to accepting any form of data flow and any application not accepting any data flow is coined to be “closed”.

Port knocking protocols are used to control the traffic flow of the data to the applications; port knocking is analogous to the prohibition period where certain sequence of knocks will allow you to enter a pub and the “secret knocks” will determine whether you’re a trustworthy source or not. The “secret knocks” in port knocking is the sequence of port knocks to the server in which if they were the correct sequence, then the server would open up any connections and listen for any form of data flow. Similar to firewalls, except the fact that port knocking programmatically doesn’t store the previous failed attempts, the packet being sent to the server contains the source IP address encrypted within the packet and the port number as well. If a sequence of port numbers can be constructed to encode some desired information, it will be possible to send that information to a machine which initially may not even appear to exist on the network.

2.0 Protocol

1. **Service to be provided**

   The service to be provided is a hidden (or invisible) web server that is visible only after a valid port knocking protocol has been completed by a client.

2. **Assumptions about the environment**

   The environment is the Internet where the web server is assumed to be running on a reachable (so, fixed IP address) host. The host supports the full set of Internet protocols.

3. Vocabulary and Encoding of Message

3.0 Port Knocking Client

The implementation of the client and the server is done in C program . The language was chosen to ensure that we maintained a steady flow of the algorithms regarding the forwarding of the packets. The first component of designing an effective port knocking system and protocol was creating the client. Port knocking is a type of communication in which information goes along closed ports. The client will be able to communicate with the server and send requests to open up a certain port. The client sends to the server a sequence of port knocks to ensure that nobody that be pretentious about the packet interception could open up the same port that the client is trying to open up since it could lead to a potential DOS (Denial of Service) attack or an untrusted source could have access to certain data coming in or out of the particular port. The sequence of knocks in our program were generated by creating a while loop to create the sockets and store the sequence into an array, We then use a source-IP based encryption i.e XORing them using a hex key value (0x55) and then sending them to the server where they are decrypted using

the same algorithm.

```
//encryptsequence(input,output);

int i;
for(i = 0; i < MAX_NUM_PORT; i++)
{
    knock_seq[i] = input[i] ^ key;
}
count = 0;
printf("\n");
printf("Ports: \n");
```

this algorithm will use the Client IP and the port number the client wants to connect to and an action either to open or close the port generates an encrypted sequence which is then sent to the Server side. This makes our protocol secure against playback effect in the sense that the client's IP and Port number are used to generate a knock sequence using a special key that is shared by both the client and server i.e if the server does not have the right key then they decrypted sequence will not match and any attacker trying to connect will be denied access without knowing what went wrong

Port Knocking and Web Server

The second primary component of the system and protocol is the port knocking server that is able to retrieve the port knocking sequence and verify that it's being passed from a trusty client. The server does this by using a decryption algorithm and the shared key to decrypt the knock sequence received in order to give remote access to the hidden web server to the IP address that was received and decrypted.The server will not give access to the Client IP until the full knock sequence is received completely and is decrypted using the same key . The server is

able to determine what the IP address of the client is and displays it. The data from the host is being stored into a structure data type, storing vital information such as the IP address, port requested to be connected, sequence of encrypted knocks, and the number of knocks received.

4. **Procedure of rules**

We first make sure that both the client and server program are installed in separate machines, we then compile and run the Server program in order to wait for a connection request, we then run the client program.In the client program we ask the user to enter the following the IP of the source host, IP of the destination host, the port number and the action to whether open or close the port and the hex key to be encrypted on Once the user enters this information the encryption algorithm and the hex key will generate a port knock sequence which is unique to the client and the port number the client is trying to connect to the server. During this process we are still assure that the Web server is still hidden. Once the sequence is complete it is sent to the port knock server where it is decrypted using the same hex key to validate that knock sequence was in the right format if either the user of client enter the wrong hex key no services will be exchanged.As mentioned earlier this security measure protects the server against a replay attack, The port knock server will spin off the local web server after verifying the sequence and clients IP and gives it remote access to the local web server in order to access files. This can be done by going to a Web browser on the client computer nd entering the IP address of the server and port number that the local webserver is running on. Another security measure taken is adding a time constraints on how long a remote host can be connected to the web server in this case we had a

timer set to 10 seconds.Setting a time makes it harder for attackers to sniff large size packets from the server as well as protect the server from a DOS attack.

Extra Credit Part

Web server could still discovered and attacked even if you are trying your best to hide it and make it unattackable . The reason why this happens because a hacker may attempt to get data from a certain site , where on the other hand a pro-hacker may cause huge damage by either defacing the webpage or use the actual web server to spread a virus. In addition , web attacks range from Layer 2 to Layer 7, which gives web server a greater chance to suffer due to the variety of possible hacking attempts. An example of a possible attack is ,Denial of service because every website is hosted on an IP address which is open to internet, and attempt of being attacked is likely to happen and make web server down.