DUKE UNIVERSITY

CS 514 / ECE 558: Computer Networks
Homework 1

Instructors: Bruce Maggs and Shane Zhang

Due: **September 13, 2021, 11:00pm**

INSTRUCTIONS: **Please use Gradescope to hand in this assignment.**
Thank you for enrolling in CS 514 / ECE 558! The aim of this assignment is to give you hands-on experience with tools for generating certificates and capturing packets, to deepen your understanding of how the SSL/TLS and SSH protocols work (or don't work properly, in the case of the Heartbleed bug), and to explore the difference between password-based and key-based authentication.
For each question, write down (or capture) all the steps you perform, together with appropriate output. We apologize in advance if some of these procedures seem tedious. Please consult the lecture slides for detailed instructions on how to generate keys and certificates.
You are free to use any platform, tools, and commands except online services, unless the problem description explicitly allows the use of online services. For example, one of the questions asks you to generate a self-signed certificate. You may **not** use a website like https://www.selfsignedcertificate.com/ which automatically generates a self-signed certificate for a given domain. You may, however, use tools like Wireshark (https://www.wireshark.org/), the OpenSSL toolkit (http://www.openssl.org/),
and the OpenSSH toolkit, (https://www.openssh.com.)
You must perform all the steps yourself, although you may rely on classmates and Ed for help.
Your answers must be submitted in a **single PDF file** (please name it your_netid.pdf) together with all the details (steps, output, etc.). Although LaTeX is preferable for formatting, you are free to use any text editor to write your answers. Feel free to contact us if you have any doubt, or, even better, post your question on Ed for the benefit of everyone!

QUESTIONS:

Here is the typical topology (i.e., figure 0.1) of a home network. You get connected to the Internet with a Modem provided by your network provider (e.g., Spectrum, AT&T, or Google). Then a home router needs to be connected to the modem, and then provide the Internet to all your devices.

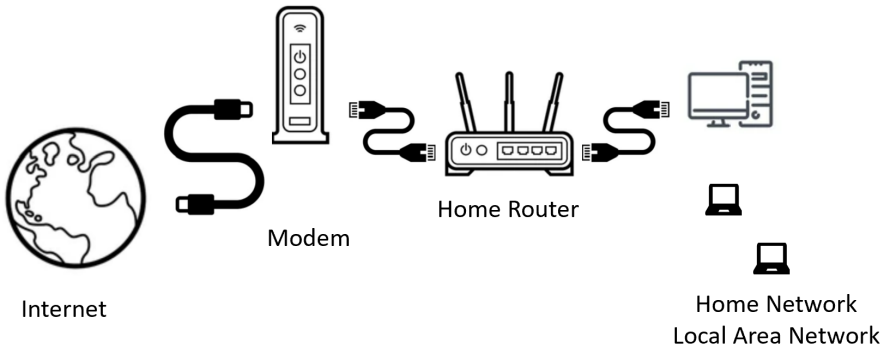

Figure 0.1: Home Network Topology

Usually, the home router can be logged in to configure the SSID, check connected devices, or even attach additional storage.
You can answer the following questions with your own home router. Or if you can not log into your own home router, you can use the following link to a home router simulator https://demoui.asus.com/, and provide screenshots if necessary.

1. What is your home router's WiFi name (SSID)?
2. What is the router's WAN(Wide Area Network) IP?
3. What is the router's LAN(Local Area Network) IP prefix, mask, mask length, range, gateway?
4. How many devices are in your home network?
5. How do those devices connect to this home router?
6. This router supports port forwarding, which can expose a specific port of a local device to the WAN IP. How can this feature be used?
7. If you browse Professor Bruce Maggs's website https://users.cs.duke.edu/~bmm/ from a local device, what IP did you reach? (hits: use ping)
8. Use the IP address you reached in the previous question to show how NAT (Network Address Translation) works here for the request from this local device to the web server.

9. How does NAT save IPv4 addresses?

10. Bob would like to log in to a server remotely, and currently does so using the SSH protocol, typing the password to his user account to authenticate. The administrator of the server has told Bob that password-based authentication will soon be phased out in favor of public-private key authentication. One reason the server administrator might have decided to phase out password-based authentication is that if the server is secretly compromised (pwned), there might be fewer repercussions. Why would this be true? (Answer was discussed in lecture.)

11. Show the steps that Bob would perform to create a suitable public-private key pair for SSH authentication. Turn in the public key that you have generated. (Don't turn in the private key!)

12. Where would Bob store the public and private keys in order to access the remote server from his laptop?

13. Sometimes when a user connects to an SSH server he/she receives a message indicating that the server's public key has changed. This message might indicate something mundane, such as a routine change of the public key by an administrator, or it might be a sign of something more serious, such as a person-in-the-middle attack. A Duke student who has not taken "CS 514 / ECE 558" always ignores these messages on the grounds that there's not much risk because the student doesn't use password authentication to log in to the SSH server, but instead uses public-key authentication. But this sense of security is false. Once the student from the previous problem has been tricked into thinking that he/she has logged in to the desired server, what might the attacker hope to obtain from the student? (The answer was discussed in lecture.)

14. Today's life lesson: Using the same public-private key pair for both authentication and encryption is risky. In this problem we will explore the risk. Here is how RSA authentication in SSH version 1.5 was described in an Internet-Draft written in November, 1995.

    > The idea behind RSA authentication is that the server recognizes the public key offered by the client, generates a random challenge, and encrypts the challenge with the public key. The client must then prove that it has the corresponding private key by decrypting the challenge.

    Suppose that you have released your public key so that people can use it to send encrypted messages to you, and you have also installed this same public key on a computer running an SSH version 1.5 server that you like to login to using RSA authentication. Eve has been eavesdropping on your communications and has captured a short email message that was encrypted with your public key and then sent to you. Eve is also the operator of the SSH server. How can Eve trick you into decrypting the message the next time you try to authenticate?

15. One way to avoid the attack described in the previous would be to use different public keys for receiving encrypted messages and for authenticating. But most people (and companies) cannot be counted on to manage their keys properly.

    As RFC 4252, from January 2006, explains, in SSH version 2, the client instead authenticates by signing a block of data that includes the session identifier, user name, and user's public key.

    > When the server receives this message, it MUST check whether the supplied key is acceptable for authentication, and if so, it MUST check whether the signature is correct.

    Suppose that you are still living dangerously and using the same public key for authentication and for receiving encrypted messages. To authenticate to an SSH version 2 server, you sign the block of data using your private key, which involves performing the same exponentiation operation using the same private key as when decrypting a message that has been encrypted with your public key. How is SSH version 2 providing better protection for you than SSH version 1.5? (The answer was discussed in lecture.)

16. You were asked to enter a "passphrase" when you created the authentication key. Bob is lazy, and decided not to provide a passphrase so that he doesn't have to type one each time he logs in to the remote server. Perhaps this isn't such a good idea. What sort of attack might Bob leave himself open to by not using a passphrase?

17. Make up a domain name (e.g., www.ilovenetworking.com) and create a certificate signing request (CSR) (e.g., web.csr.pem) for that domain. Write down all the steps and show the CSR you generated. We will provide a file called openssl.conf on Sakai for use with the openssl command-line tool.

18. Next, you need to have the CSR signed by a certificate authority (CA). For this assignment, you will act as the CA. So you'll need to start by creating a certificate for a "root" CA. First create a root key (e.g., ca.key.pem), and then use the root key to create a root certificate (e.g., ca.cert.pem). Write down all the steps and show the certificate you generated.

19. Normally the root certificate authority does not directly sign certificates for domains. Instead, they are signed by intermediate certificate authorities. So first create an intermediate key (e.g., intermediate.key.pem), then use that to create a CSR (e.g., intermediate.csr.pem). Finally, use the root key to sign the CSR, resulting in a signed intermediate certificate (intermediate.cert.pem). Write down all the steps and show the certificate you generated.

20. Next, you will use the intermediate key to sign your web certificate (web.csr.pem). Let's call this signed certificate web.cert.pem. Write down all the steps and show the certificate you generated.

21. Assume you are using web.cert.pem as a certificate for your Web site. When you try to make a secure connection to your Web site what message will be displayed on your browser? Why? [Note, you don't have to set up an actual Web site, nor do you have to actually use this certificate to answer this question].

22. Sometimes when you visit a web site, your browser receives multiple certificates. As an example, use a packet trace to determine which certificates are received when visiting https://www.cs.duke.edu. Note that because of a recently introduced mechanism called TLS session caching, a web server may only send your browser certificates on the first HTTPS connection.

23. Why would a web site send more than one certificate to a browser, and what does the browser do with the certificates once it receives them? (The answer was discussed in lecture.)

24. What certificate authorities are recognized by your browser? List any 5 of them.

25. Select a message of your choice. Create a hash of the message using the MD5 message digest (i.e., hash) algorithm. You may use an on-line tool to compute the MD5 message digest, if you'd like. Sign the hash. Finally, verify the signed message. Note that in order to sign and verify the message, you will need to create public-private key pairs.

26. Legacy protocols often send user names and passwords over the network in the clear! For this problem, you will log in to an ftp server whose network address will be posted on Ed. The user name is **ftp_test** and the password is **cps514**. Demonstrate that the password is sent in clear text by providing a snapshot of a packet trace using WireShark or other appropriate tool. Note: This exercise is only for understanding the security flaw with the ftp protocol. In practice, you should always use the secure version of ftp (SFTP) or some other secure file transfer protocol such as SCP. Also note that we are only asking you to capture your own packets and the password to this dummy account. Capturing the packets and passwords of other users is not encouraged and may not be legal.

27. Your loyal instructors have set up a web server where you can enter your user name and password. (Don't enter a real password!) The address of the server will be revealed on Ed. Alas, the version of openssl on the server is vulernable to the Heartbleed bug. Your job is to show that by exploiting Heartbleed, you can steal user names and passwords from the server. The server is not available to all Internet users – you may find that you can access it only from certain network addresses at Duke. More details will be posted on Ed if necessary.

    We don't expect you to write a Heartbleed exploit toolkit from scratch. Instead, you can use the tools available at https://github.com/sensepost/heartbleed-poc to probe the server.

28. We will provide you with the name of a Web server that includes a revoked certificate in its “server hello” message when it receives a request for a TLS connection. Your job is to demonstrate that at least one browser detects that the certificate has been revoked, while another does not.