Name: Nicholas Mason

Course: CS445 – Internet Security

Date: April 28th, 2018

CS 445 – Internet Security Technical Report

Introduction

SSL certificates

SSL certificates were originally created in order to provide an additional layer of security when a private party is attempting to communicate on the internet, specifically connecting to websites where confidential information may be passed between each party. A website where confidential information is needed to complete a transaction or process, involving such information as a user’s name, password, banking information, or social security number, must now have a SSL certificate. Having such a certificate allows the website to obtain a distinct public and private encryption key, further providing a higher level of security online.

A SSL certificate can be obtained by a given website by going through the process of finding a 3rd party Certificate Authority, also called a CA. The CA will possess the ability to authenticate and legitimize the intent of the website. If the website passes the qualifications of the CA, the website will gain access to their own unique public and private encryption keys for their specific IP address or domain name of their website. To a user on the Internet, the lock icon next to the URL and the URL starting with the symbols HTTPS ensures the user that the website has obtained some sort of SSL certificate from a Certificate Authority, resulting in their personal information being encrypted.

Creating a Self-Signed SSL certificate

Creating an SSL certificate creates many benefits for the person creating the SSL certificate, or for the website or server in which the certificate is made for. This process can help a website or server in fighting against various cyber-attacks. One of the most recognized attacks that can be avoided by creating a self-signed SSL certificate is the “man in the middle” attack. A man in the middle attack can be depicted as a user sitting between two party, listening to the content or traffic between each party. While listening to the traffic between each party, the malicious user can potentially relay or alter the communication between the two parties, resulting in relaying false information. Through the process of creating a self-signed SSL certificate, a server or website of any kind is encrypted, hindering the malicious users ability to listen to the traffic between various groups or parties.

In order to successfully create an SSL certificate on a Linux machine, which is convenient for this project, various steps must be completed in order to ensure that the encryption will be executed correctly. To start, creating an RSA keypair is required. Using openSSL in the Linux environment will allow for built in functionalities to assist in this process. An RSA key pair consists of two types of keys, private key and a public key. The RSA can be defined as “private key based on RSA algorithm. Private Key is used for authentication and a symmetric key exchange during establishment of an SSL/TLS session” (Namecheap, Inc.). This is one of the parameters needed when creating a SSL certificate. The public key will look that there exists asymmetric encryption, meaning that there is a private key and a public key included in the SSL certificate. The private key will not be present to any person in this process because it is unique to the server or website that the certificate is being made for. In the case of a self-signed certificate, the private key can be extracted to a folder because the user creating the SSL certificate is also the user who owns the domain or server that is being encrypted.

Upon extraction of the private key into a local directory on the Linux machine, replacing the IP address of the server, or simply typing the domain name of the website in which the user is creating the certificate for must be performed. With the newly created key in the proper location, a CSR file, also called a certificate signing request, is created and signed by the user, or sent off to a certificate authority for inspection and verification of authenticity. OpenSSL handles the logistics of what information is needed in order for the certificate signing request to be properly handled, resulting in the CSR file being successfully created. Because the CSR file has been successfully created and placed into a desired directory, the creation of the certificate, or the .crt file, can finally be performed. This finally allows the user to confirm that the certificate creation has been successful. By opening a simple Apache HTTP server, and changing the certificate file and key file to the newly created and authenticated files, the user is able to see a valid certificate for the HTTP server. Although the certificate is not in the trusted CAS list stored by a valid web browser, the HTTP server is encrypted, allowing the information contained on the server to be more secure. Unlike a certificate authority issued certificate, self-signed certificates are not verified by a trusted certificate authority, resulting in the user seeing a security alert message prior to visiting a website on a popular web browser.

Conclusion

References

<https://www.namecheap.com/support/knowledgebase/article.aspx/798/67/what-is-an-rsa-key-used-for>