Assessment 4 - Report

COSC540 - Networks and Information Security

Andrew Weymes

2023-05-04

Introduction

The previous version of the client-server model suffered a number of security, privacy, and trust issues due to the lack of encryption and network security. Using the CIA triad, these issues are listed below, along with approaches to mitigating them:

Confidentiality

Without encryption, anyone with access to the network could easily intercept and read message being sent between the client and server. This means that any sensitive information sent over the network, such as usernames, passwords, or credit card numbers, could be stolen.

To mitigate this issue, the approach was to use encryption to protect the confidentiality of the data being sent between the client and server. Specifically, OpenSSL was used to encrypt the communication between the client and server. This should ensure that the data being sent between the client and server is protected from eavesdropping by third parties.

A limitation is that encryption alone is not foolproof and can be vulnerable to attacks like man-in-the-middle attacks, in which an attacker can intercept and alter the messages between the client and server. To mitigate this risk, authentication should be implemented to ensure that the client and server are who they say they are, which will be addressed in the next point.

Integrity

Without any form of integrity protection, it is possible for an attacker to tamper with the data being sent between the client and server, leading to data corruption or loss.

To mitigate this issue, the approach should be to use message authentication codes to ensure the integrity of the data being sent between the client and server. Specifically, a hash function like SHA256 could be used to generate fixed size a MAC for each message being sent, which would be included with the message. The server could then verify the MAC to ensure that the message has not been tampered with. This current implementation does not use this feature due to the difficulties in understanding the esoteric documentation of the OpenSSL library in addition to the lack of any easy to follow or up to date demonstrations.

The limitation of this approach is that SHA256 only ensures message integrity and does not protect against other types of attacks, such as replay attacks or denial-of-service attacks. Furthermore, hashing functions should always be reviewed to ensure they have not been cracked or substantially weakened by techniques such as rainbow tables.

Availability

Without any form of network security, it is possible for an attacker to launch denial-of-service attacks against the server, which can cause the server to become unavailable to legitimate users.

To mitigate this issue, an approach could be to use measures such as firewalls and intrusion detection systems to protect the server from DoS attacks. Load balancing could also be used to distribute load across multiple servers, helping to prevent any single server from becoming overloaded and unavailable. This mitigation was not implemented as it is out of the assessment scope.

Some limitations of this approach is that it can be difficult and expensive to implement and manage network security measures, and they may or may not be effective against sophisticated attacks. Additionally, load balancing can require significant resources to implement and maintain, and can also open up additional risks depending on the geolocation and local laws.

Authentication

While not specifically part of the CIA triad, a lack of authentication makes it possible for an attacker to impersonate the server or client, leading to unauthorized access. This means that an attacker could potentially steal sensitive information or even take control of the system, depending on the client's authority.

To mitigate this issue, the proper approach to use should be the exchange of digital certificates to authenticate the server and the client. Specifically, using the public key infrastructure to issue digital certificates to the server and the client, which would be used to authenticate them to each other during the TLS handshake. This would ensure that the server and client are who they claim to be, which would prevent unauthorized access to the system.

The limitation of this approach is that PKI requires a trusted third party, known as a certificate authority, to issue and manage the digital certificates. If the authority is compromised, then the entire PKI system could be vulnerable to attack. Additionally, PKI can be complex to implement and manage, and can require significant resources to maintain. Further to this, the current implementation of this assignment uses a self signed certificate on the server end only with no validation at the client end. Again, this was not done as getting a validated certificate is certainly out of scope for this assignment. Additionally, I felt that this implementation was satisfactory to demonstrate my knowledge of signed digital certificates.