20.07.2020

GUARDIAN

SECURITY SPECIFICATIONS

DEKOB
— TECHNOLOGY —

CONTENTS

DEKOB
— TECHNOLOGY —

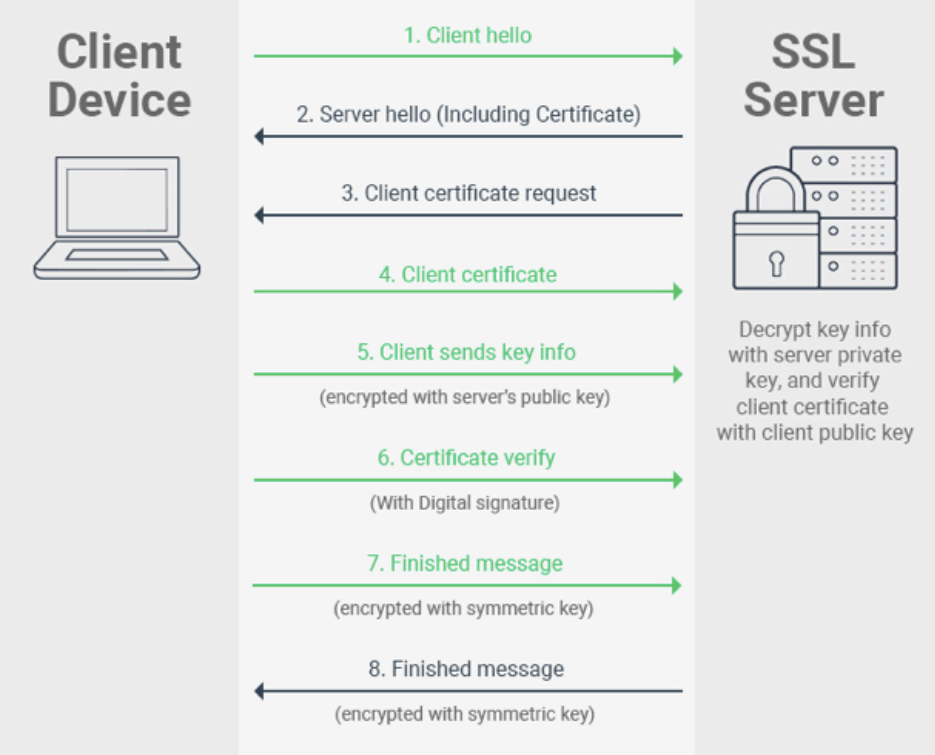
1. GENERAL INFORMATION

APIs, applications and websites are key channels for doing business with customers and suppliers. As more and more shift online, ensuring these resources are secure, performant and reliable is a business mandatory.

Keeping an internet connection secure and safeguarding any sensitive data that is being sent between two systems, preventing criminals from reading and modifying any information transferred, including potential personal details is a challenging staff. The two systems can be a server and a client or server to server. For Guardian these two systems can be listed like; server and as clients web/mobile apps.

1.1. Transport Layer Security

TLS is a widely adopted security protocol designed to facilitate privacy and data security for communications over the Internet. TLS encryption can help protect applications from attacks such as data breaches (brute force, man-in-the-middle) and DDoS attacks.



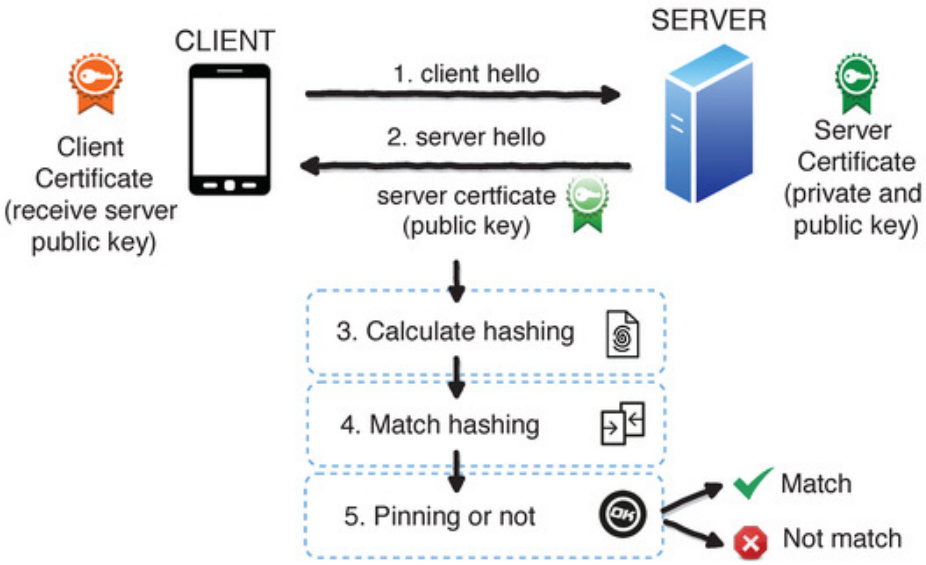
Guardian takes security issues seriously, starting with the authentication phase. With the help of TLS abilities both web application and mobile apps are secured especially for man in the middle attacks. Web application is being served over HTTPS which is an implementation of TLS encryption on top of the HTTP protocol. TLS pinning is implemented for mobile apps which detailed in section 1.2.

Guardian “User Ban Mechanism” has been designed to prevent unwanted vulnerabilities like brute force attacks. The system stores User IDs to the origin server which has many failed attempts for authentication. By temporarily blocking, entry of these IDs is prevented by this system. Block duration time can be set according to customer demand and needs.

In simple terms DDoS attack means that multiple computers send fake requests to the target in larger quantity. The target is flooded with such requests, thereby the resources become unavailable to legitimate requests or users, which is undesirable. Guardian can work compatible with any DDoS prevention tool in the market.

1.2. TLS Certificate Pinning

For mobile applications, TLS certificate pinning goes a long way toward building security into an app and enhancing user and data privacy. It provides privacy and data integrity between two communicating entities, such as a mobile app and a backend API(server).

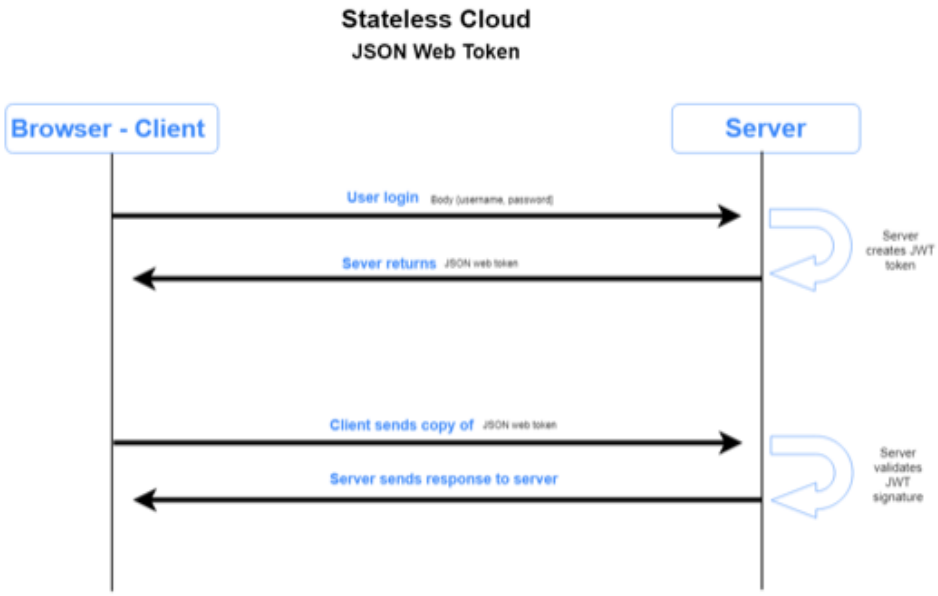


TLS pinning increases the cost to attack the mobile app as it's very hard to break the encryption that shields it from malicious packets sent through the encrypted channel. With TLS pinning, app data is encrypted across the network and does not allow third party inspection. Thus, as mentioned above, Guardian mobile apps eliminate the possibility of man in the middle attacks and keeps the user safe.

2. JSON Web Tokens

Token-based authentication enables users to obtain a token that allows them to access a service and/or fetch a specific resource without using their username and password to authenticate every request. The authentication token is created by the authenticating service and contains information to identify a particular user and the token validity. The token itself is cryptographically signed to prevent tampering.

JSON Web Token (JWT) defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed. JWTs can be signed using a secret or a public/private key pair.

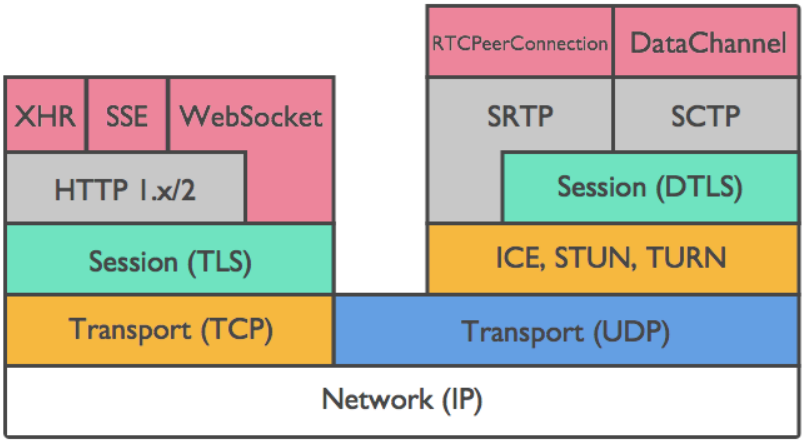


On server side, Guardian also takes the advantage of JWT instead creating a unique hash for a client, store it in the database and verify that hash against the incoming requests every single time. In Guardian we use RSA algorithm functionalities, to hash related information in JSON object with a secret, for creating the tokens. Guardian does not save JWTs in database, stores them on client which increases the performance.

3. WebRTC

WebRTC enables direct media-rich communication between two peers, using a peer-to-peer (P2P) topology. WebRTC resides within the user's browser, and requires no additional software to operate. For native clients, like Android and iOS applications, a library is available that provides the same functionality. The actual communication between peers is prefaced by an exchange of metadata, termed "signalling". This process is used to initiate and advertise calls, and facilitates connection establishment between unfamiliar parties.

Here is the protocol stack of the WebRTC:



DEKOB
— TECHNOLOGY —

Guardian involves TURN, DTLS and SRTP protocols listed in the stack above. In the eventuality that establishing a P2P communication fails, a fallback option can be provided via a TURN server. WebRTC uses DTLS-SRTP to add encryption, message authentication and integrity, and replay attack protection. It provides confidentiality by encrypting the RTP payload and supporting origin authentication.

DTLS is used for encrypting data streams, while SRTP is used for encrypting media streams. However, DTLS-SRTP is typically used to perform the SRTP key exchange to detect any man-in-the-middle attacks. SRTP is one component of this security in WebRTC. It gives comfort to developers looking for a reliable and secure API. SRTP uses Advanced Encryption Standard (AES) as the default cipher, in Guardian AES-256 is used.