8.3.7 User to Native Airport application:

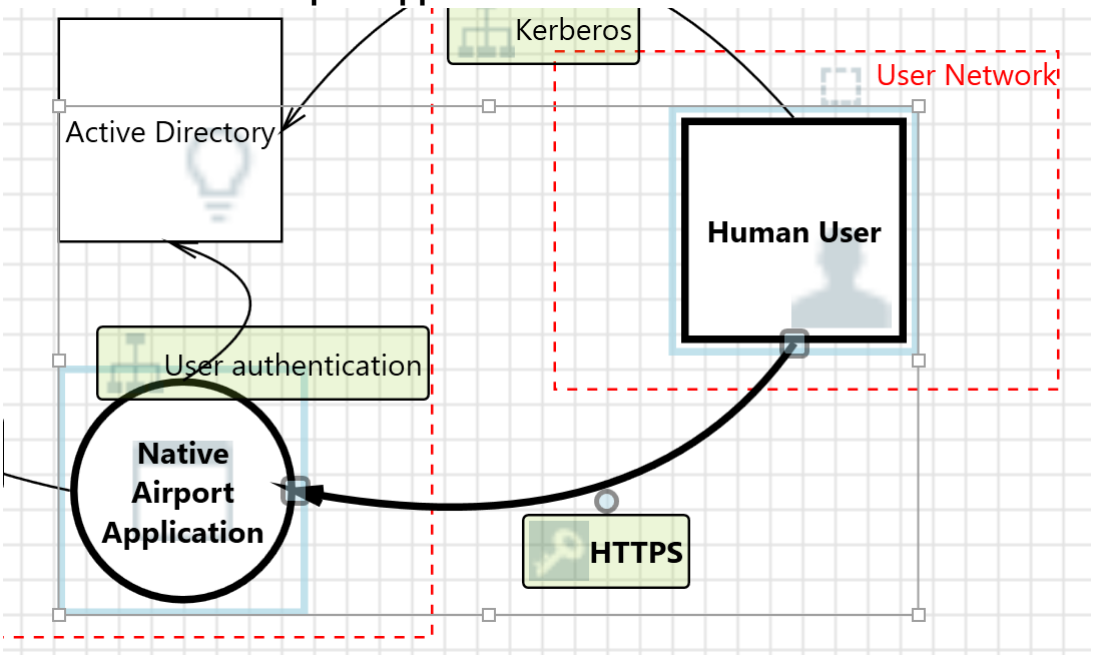

Figure 12: User to Native application

Figure 12 the interaction between the user and the native application is shown.

Listed below are potential threats and corresponding measures to mitigate them.

Spoofing the Human User External Entity

Category:	Spoofing
Description:	Human User may be spoofed by an attacker and this may lead to unauthorised access to Native Airport Application. Consider using a standard authentication mechanism to identify the external entity.
Mitigation:	User authentication will be done through Single Sign On (SSO) with MFA enabled for all external connections (Grassi et al., 2017).

Potential Data Repudiation by Native Airport Application

Category:	Repudiation
Description:	Native Airport Application claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.
Mitigation:	WAF will protect web server access. WAF will provide details of all the access to the application, including time of access, IP address and resource accesses.

Elevation Using Impersonation

Category:	Elevation Of Privilege
Description:	Native Airport Application may be able to impersonate the context of Human User in order to gain additional privilege.
Mitigation:	User authentication will be done through Single Sign On (SSO) with MFA enabled for all external connections (Grassi et al., 2017).

Cross Site Request Forgery

Category:	Elevation Of Privilege
Description:	Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker forces a user's browser to make a forged request to a vulnerable site by exploiting an existing trust relationship between the browser and the vulnerable web site. In a simple scenario, a user is logged in to web site A using a cookie as a credential. The other browses to web site B. Web site B returns a page with a hidden form that posts to web site A. Since the browser will carry the user's cookie to web site A, web site B now can take any action on web site A, for example, adding an admin to an account. The attack can be used to exploit any requests that the browser automatically authenticates, e.g. by session cookie, integrated authentication, IP whitelisting. The attack can be carried out in many ways such as by luring the victim to a site under control of the attacker, getting the user to click a link in a phishing email, or hacking a reputable web site that the victim will visit. The issue can only be resolved on the server side by requiring that all authenticated state-changing requests include an additional piece of secret payload (canary or CSRF token) which is known only to the legitimate web site and the browser and which is protected in transit through SSL/TLS. See the Forgery Protection property on the flow stencil for a list of mitigations.
Mitigation:	Access to Native application will be protected with Web Application Firewall. A WAF can inspect the input data coming from user requests and apply strict validation rules. It checks for any suspicious or malicious content, such as JavaScript code or HTML tags, and blocks or sanitises them to prevent CSRF attacks.