alpha Software®

Alpha Anywhere

Persistent Login in a Mobile Application

The Requirement

- User should only have to login to the Alpha Anywhere Server once.
- Subsequent interactions with the server should not require the user to login again – even if the server session has expired. (Unless permission has been revoked).

The Solution

- JSON Web Tokens (JWT) – stored in LocalStorage on the device

The Solution (in detail).

- User launches a Mobile app.
- User has never logged into Alpha Anywhere server before.
- Mobile app presents a login screen to the user.
- User enters username/password and clicks Login button.

The Solution (continued).

- Custom Ajax callback is made to the server to log user in.
- If user logs in successfully, a JWT is created on the server.
- JWT encodes the username and password and an optional expiry data.
- Ajax callback sends a Javascript response to store the JWT in localStorage and also in the UX's state object (so that the JWT will be submitted on every subsequent Ajax callback).

The Solution (continued).

- Code in the UX's client-side onRenderComplete event does the following:
  - Check if the token is in LocalStorage and is not blank
  - If so, give focus to the PanelCard for the home screen of the App (bypassing the Login screen). Optionally first give focus to a client-side login screen.
  - Set the token into the UX's state object so that it will be submitted to the server on all Ajax callbacks
  - If token not found, or is blank, give focus to the Login Panel so that user can log into the App.

What happens when a user makes an Ajax Callback?

- Server-side canAjaxCallback event will fire.
- Permission for callbacks to login or logout are automatically authorized. Permission to execute all other callbacks must be determined as follows:
- Name of current logged in user is retrieved from the security framework.
- If current logged in user is not blank, user is still logged in and permission to execute the callback is granted.
- If current user name is blank, the user is no longer logged in (session may have expired).
- Decode the JWT (which is automatically submitted on all callbacks) and see if it is still valid.
- If JWT is valid, extract the username and password from the decoded JWT and log the user in again using the credentials in the JWT.
- Login can either succeed, or fail (if user's account has been terminated).
- Generate a new JWT and send it back to the client.
- Authorize the Ajax callback.

Ajax Callbacks (continued)

- If user is no longer logged in, and if the JWT has expired, send a response to the client directing them to login again.
- Permission to execute the Ajax callback is denied.

Summary

- By storing the encrypted JWT in LocalStorage the user no longer needs to log into the Alpha Anywhere server each time they start the App (or if their session expires while they are running the App)