# **Google 2FA**

# **1 Basics**

A table with an overview of the support status and applicability.

|  |  |
| --- | --- |
| Status: | **Supported** |
| Architecture(s): |  |
| Component(s): | Google authenticator, NodeJS |
| Hardware: | *Smartphone device* |

# **2 Overview**

Google offers state-of-the-art TOTP (Time Based One Time Password) authentication through their Google Authenticator mobile application. This offline authentication method is extremely secure especially in comparison to traditional email or SMS authentication models.

# **3 User details**

This optional feature can be enabled in the user security settings and set up using their smartphone device. It can be turned on as a login gate as well as a gate to many other sensitive features. G2FA can also be used as a replacement for the PIN code.

# **4 Technical details**

Google has released an entire library for their APIs available with npm: <https://github.com/google/google-api-nodejs-client/>

This should make implementing their authentication techniques very simple within the wallet and desktop applications.

# **5 Limitations**

Users do need a smartphone for this feature to work.

# **6 Testing**

Testing should be fairly simple. Implement the feature and set it up on a devops device.

# **7 Areas for improvement**

Tutorials will be necessary for users who are unfamiliar with 2FA or TOTP.

# **8 Known issues**

If a user loses their device and does not save their recovery phrase, it can be very difficult to access features or accounts locked with G2FA.

# **9 References**

<https://github.com/google/google-api-nodejs-client/>