Transaction Authorization

Introduction

Some applications use a second factor to check whether an authorized user is performing sensitive operations.

A common example is wire transfer authorization, typically used in online or mobile banking applications.

For the purpose of this document we will call that process: transaction authorization.

Transaction authorization can be implemented using various methods, e.g.:

- Cards with transaction authorization numbers (TAN)
- Time based OTP tokens, such as OATH TOTP (Time-based One-Time Password),
- OTP sent by SMS or provided by phone
- Digital signature using e.g. a smart card or a smart phone,
- Challenge-response tokens, including unconnected card readers or solutions which scan transaction data from the user's computer screen.
- Some of these can be implemented on a physical device or in a mobile application.

Functional Guidelines

- Transaction authorization method has to allow a user to identify and acknowledge significant transaction data
- Change of authorization token should be authorized using the current authorization token
- Change of authorization method should be authorized using the current authorization method
- Users should be able to easily distinguish the authentication process from the transaction authorization process
- Each transaction should be authorized using unique authorization credentials

Non-functional guidelines

- Authorization should be performed and enforced server-side
- Authorization method should be enforced server side
- Transaction verification data should be generated server-side
- Application should prevent authorization credentials brute-forcing
- Application should control which transaction state transitions are allowed
  - Transaction authorization is usually performed in multiple steps, e.g.:
    - The user enters the transaction data.
    - The user requests authorization
    - The application initializes an authorization mechanism.
    - The user verifies/confirms the transaction data.
    - The user responds with the authorization credentials.
    - The application validates authorization and executes a transaction.
- Transaction data should be protected against modification
- Confidentiality of the transaction data should be protected during any client / server communications
- When a transaction is executed, the system should check whether it was authorized
- Authorization credentials should be valid only by limited period of time
- Authorization credentials should be unique for every operation