**Simple Trust-Elevation Use Case – Online banking transactions**

Description: A bank customer initially logs on to the bank site (through a browser or mobile app) to view his/her account balance. Then, he decides to perform a higher risk transaction that requires a higher level of authentication (let's say a funds transfer of $X).

Pre-conditions:

* Claimant has an existing relationship with the bank (i.e., is an account holder)
* Claimant has previously registered his authentication methods (e.g., his password, device, biometric)
* Transaction 1 has been mapped to an LOA1 risk level and Transaction 2 has been mapped to an LOA2 risk level.

Policy Table\*:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Transaction****Risk Level** | **Initial Strength** | **Desired Strength** | **Authentication needed\*** | **Policy designation** |
| Low | LoA0 | LoA1 | One factor, either what you know or have | P1 |
| Med | LoA0 | LoA2 | Two factors, any class | P2 |
|  | LoA1 | LoA2 | One factor, different than used for LoA1 authentication | P3 |
| High | LoA0 | LoA3 | Three factors | P4 |
|  | LoA1 | LoA3 | Two factors, any class | P5 |
|  | LoA2 | LoA3 | One factor, different than used for LoA1/2 authentication | P6 |

Where LoA0 represents a "user not logged in" state.

\*Authentication policies are set by the relying party.

Method Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method designation** | **Method description** | **Class(es)** | **SF strength** | **Threats addressed\*** |
| M1 | PIN (>=4 char) | Know | 1 |  |
| M2 | Password (>=8char) | Know | 1 |  |
| M3 | Device ID | Have | 1 |  |
| M4 | Crypto key (TLS protocol) | Have | 2 |  |
| M5 | Biometric – face  | Are | NA |  |
| M6 | Biometric – fingerprint  | Are | NA |  |
| M7 | PIN + Device ID | K+H | 2 |  |
| M8 | Crypto key + face | H+A | 3 |  |

\*For benefit of relying party operators setting up policies. (Could reference other portions of document.)

Process flow:

* Claimant accesses bank site/app.
* Claimant logs onto his account (at LoA1) to check account balance
	+ Going from LoA0-1
	+ Policy P1 is invoked
	+ User enters password
		- Authentication server verifies password
	+ Access is granted, balance displayed
* Claimant then chooses to transfer funds (Med risk level)
	+ PDP determines current Authn level (LoA1)
		- Looks up current state for that user
	+ PDP determines target Authn level (LoA2)
	+ PDP demands trust elevation (LoA1-2)
		- Request sent to MD
		- MD selects Policy P3 to meet the TE requirement
		- MD returns response to PDP
	+ PDP sends Authn request to Authn server, specifying policy P3
	+ Authn server determines currently available methods
		- Authn server issues appropriate challenges to user
		- User/device responds to challenges
		- Authn server verifies Authn data
	+ Authn server send Authn results to PDP (standard assertion, e.g., SAML)
	+ PDP makes access decision and send to PEP
	+ PEP provides access to requested resource (i.e., funds transfer function)
	+ User proceeds with transfer

Content of PDP-MD request:

* Current level
* Method(s) used to achieve current level
* Target level

Content of MD-PDP response:

* List of methods that could be used to achieve target level

Content of PDP-AuthnSvr request:

* Claimant ID
* List of acceptable methods
	+ Issue – support by existing standards (SAML/OIDC)?

*Issues and Actions*

* Which component serves as the 'orchestrator'?
* Options for allocation of functions to components
* Define architecture/framework
* Identify any changes needed to existing standards
* Need for an open source reference implementation