ASP.NET Identity – Security Features

ASP.NET Identity – Security Features

ASP.NET Identity comes with several modern safety features such as:

- Two Factor Authentication
- Account Confirmation
- Password Reset
- Account Lockout
- Security Stamp
- Security Token Service

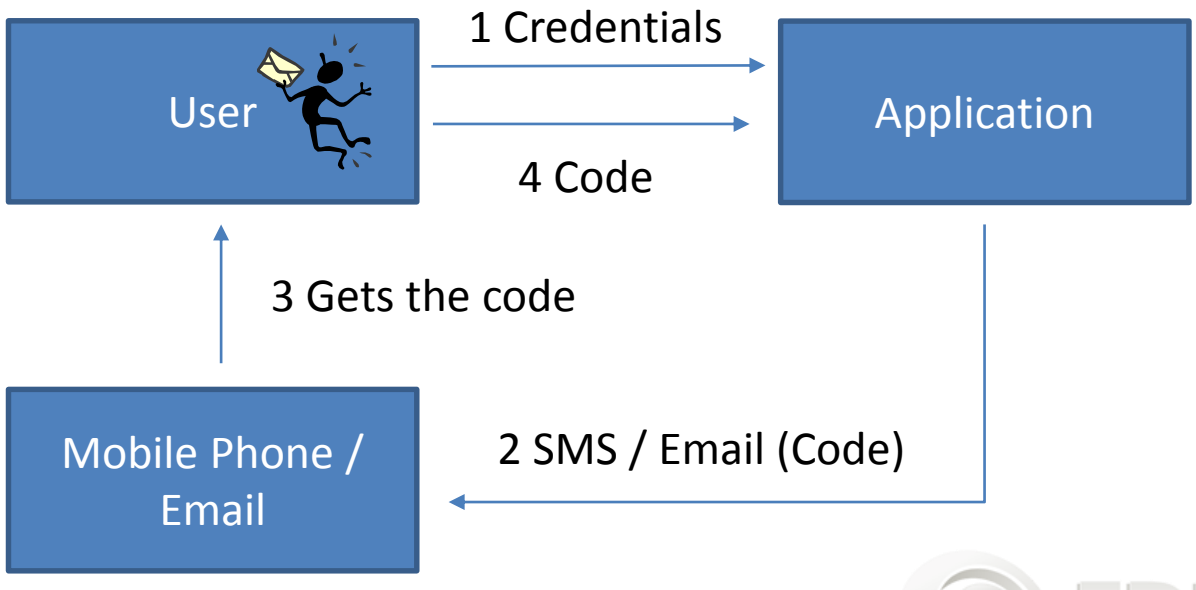
ASP.NET Identity – Security Features

Two Factor Authentication

- Requires the user to provide two different “Secrets”
- A second security layer if the users password is compromised
- Email/SMS

Two Factor Authentication

- The user presents its credentials to the application
- The application sends a code in either a text message or email to the user
- The user receives the code from the mobile phone / email
- The user presents the code to the application

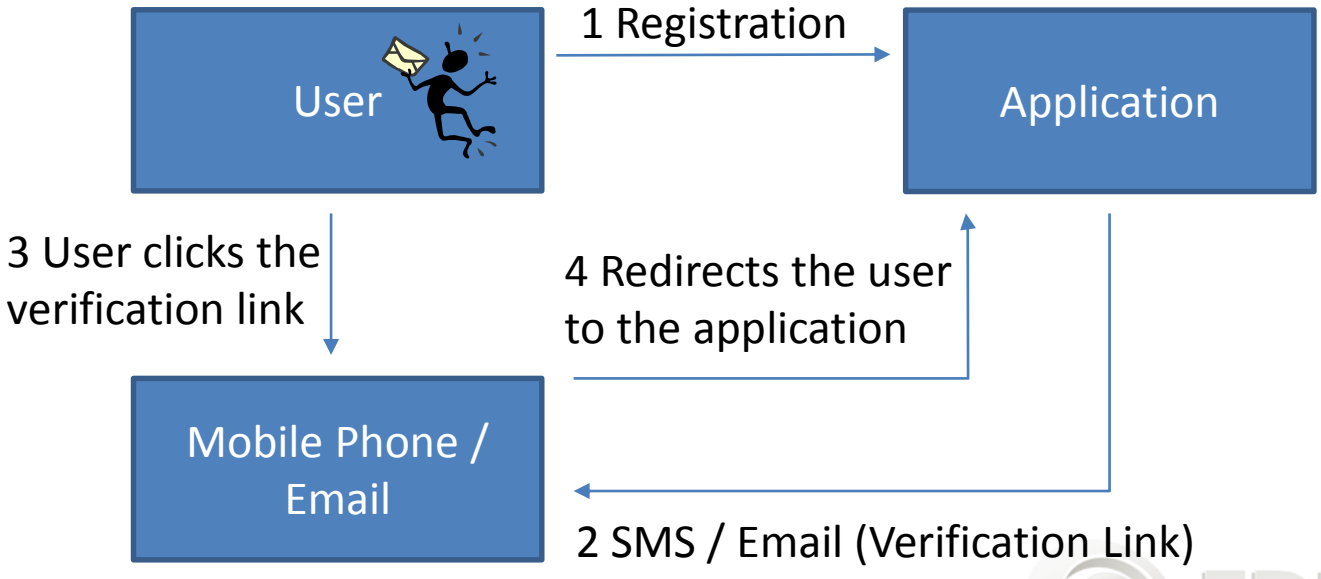


Account Confirmation

- Sends a email / SMS with a verification link
- Verification link has a one day lifespan per default
- A way to confirm that the user has access to a second “Secret”
- Useful for two factor authentication & password recovery features
- Uses a SMTP or third party service to send email / SMS

Account Confirmation

- The user registers to an application
- The application sends an SMS/ email to the email / phone number which the user entered in the registration
- The user checks his email / phone for the verification code
- The verification link redirects the user to the application



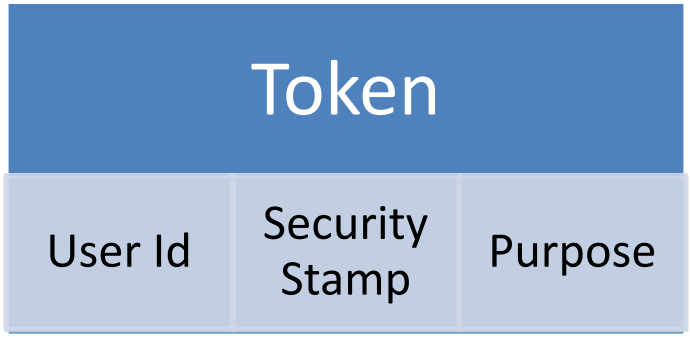
Password Reset

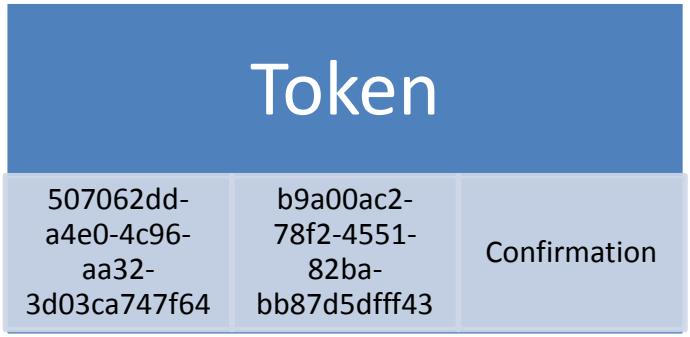
- The user can request a password reset by providing the application with a username
- Sends a email / SMS with a reset link
- Reset link has a one day lifespan per default
- Requires account confirmation

ASP.NET Identity – Security Features

Generating the link

When we generate a confirmation or reset link we first generate a token which contains the User Id, Security stamp and purpose.

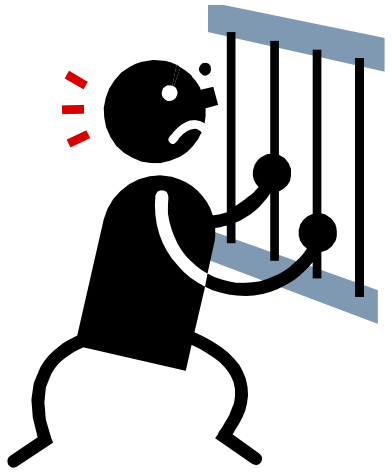




http://localhost:1470/Account/ResetPassword?userId=a8b1389c-df93-4dfc-b463-541507c1a4bc&code=yhUegXIM9SZBpPVbBtv22kg7NO7F96B8MJi9MryAadUY5XYjz8srVkS5UL8Lx%2BLPYTU6a6jhqOrzMUkkMyPbEHPY3Ul6%2B%2F0s0qQvtM%2FLLII3s29FgkcK0OnjX46Bmj9JlFCUx53rOH%2FXMacwnKDzoJ1rbrUyypZiJXloIE50Q6iPuMTUHbX9O%2B3JMZtCVXjhhsHLkTOn9IVoN6uVAOMWNQ%3D%3D

Account Lockout

- Prevents the user from logging in after a number of failed login attempts
- The lockout time is set to 5 minutes by default
- Prevents brute force attacks
- Compensates for weak password policy



ASP.NET Identity – Security Features

Security Stamp

- A value which is changed when something security related is altered
- Invalidates existing cookies & tokens if the value is changed

Ex. The security stamp is changed when the user requests a password reset, invalidating all existing cookies since the security stamp has changed. This would deny an attacker access to a possibly compromised user account.

Security Token &
Security Token Services

ASP.NET Identity – Security Features

What is a Security Token?

A Security token is a token which is used to authenticate users.

The security token contains an **ID**, **security key** as well as the time from which it is valid & for how long.

Type	Description
ID	Unique identifier
Security key	Cryptographic key
ValidFrom	Time at which the token is valid from
ValidTo	Time at which the token is valid to

What is a STS (Security Token Service)?

- A STS is a service which provides users with security tokens
- This token is then used to authenticate the user on the web-application

ASP.NET Identity – Security Features

A good example of where a STS is used is when you try to login to your online bank.

The bank requests a code (a security token) from you, which you get by entering your credentials to your authentication device (STS) which will return a code (a security token).

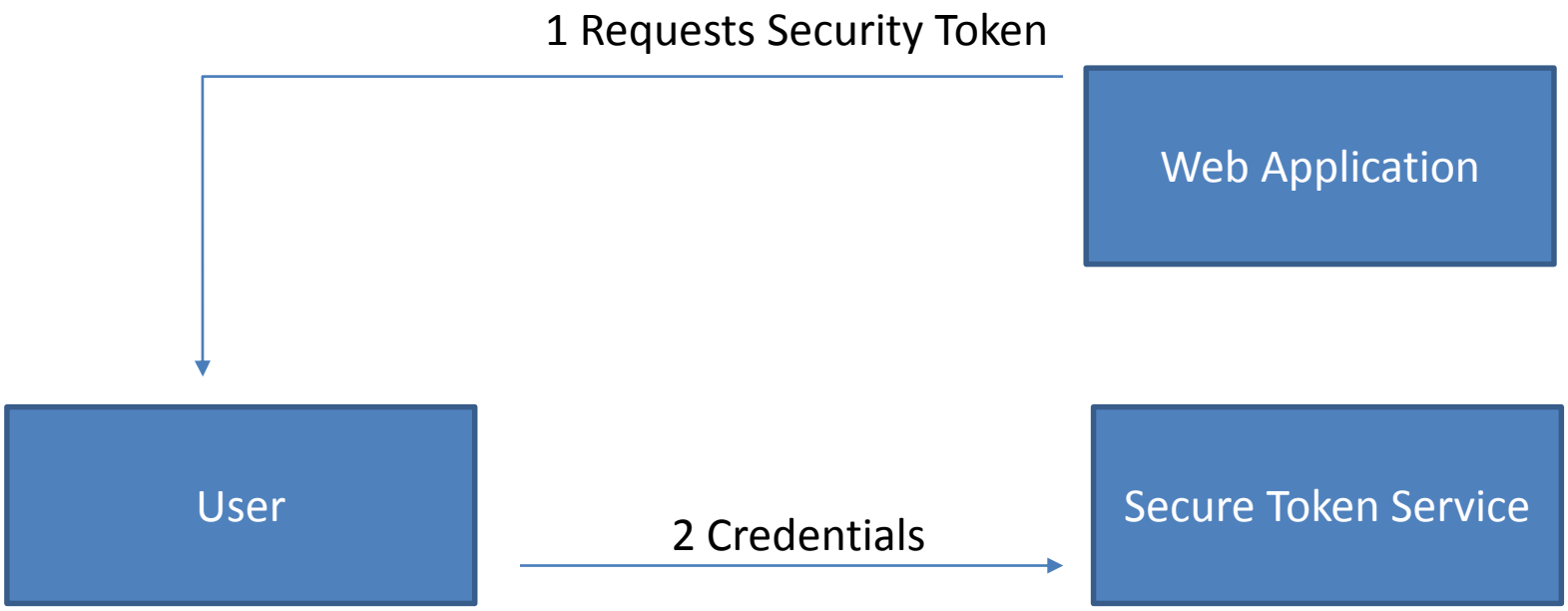
You would then proceed to use this code to login.

STS Flow illustration

When the user tries to login to the web application, the application requests a security token.

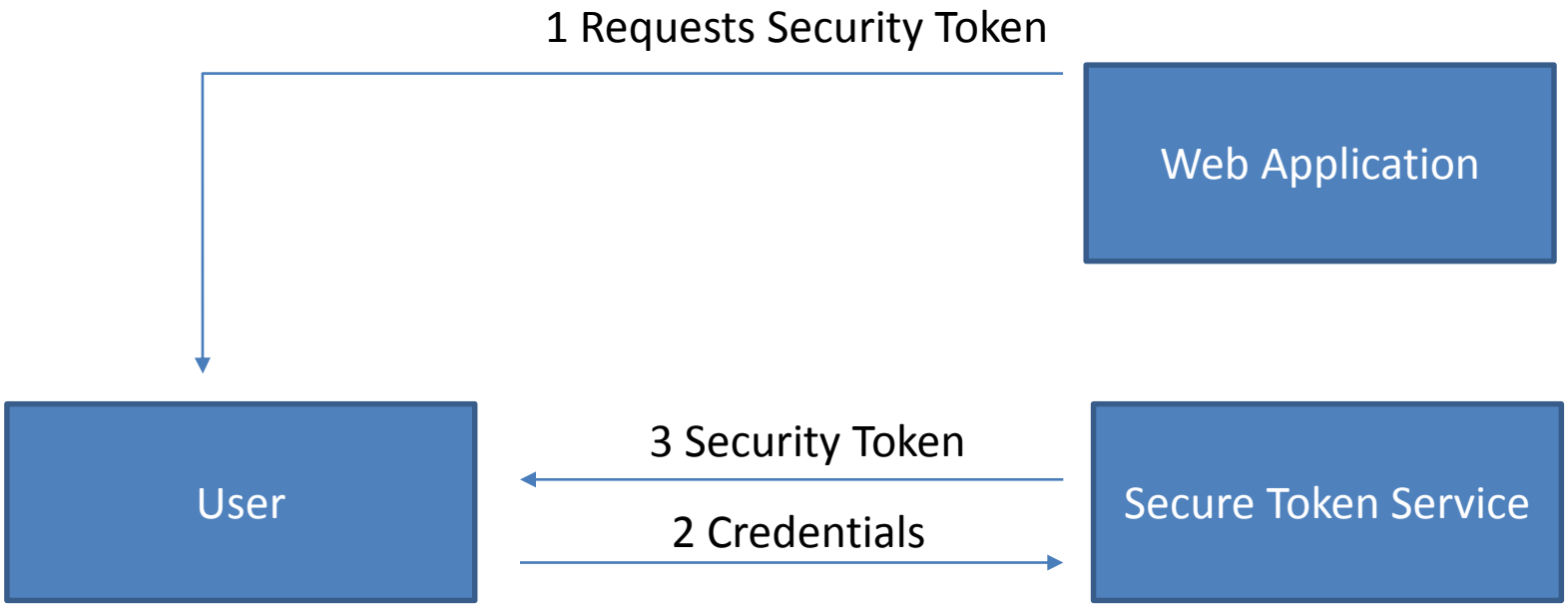
STS Flow illustration

The user provides the STS with his credentials



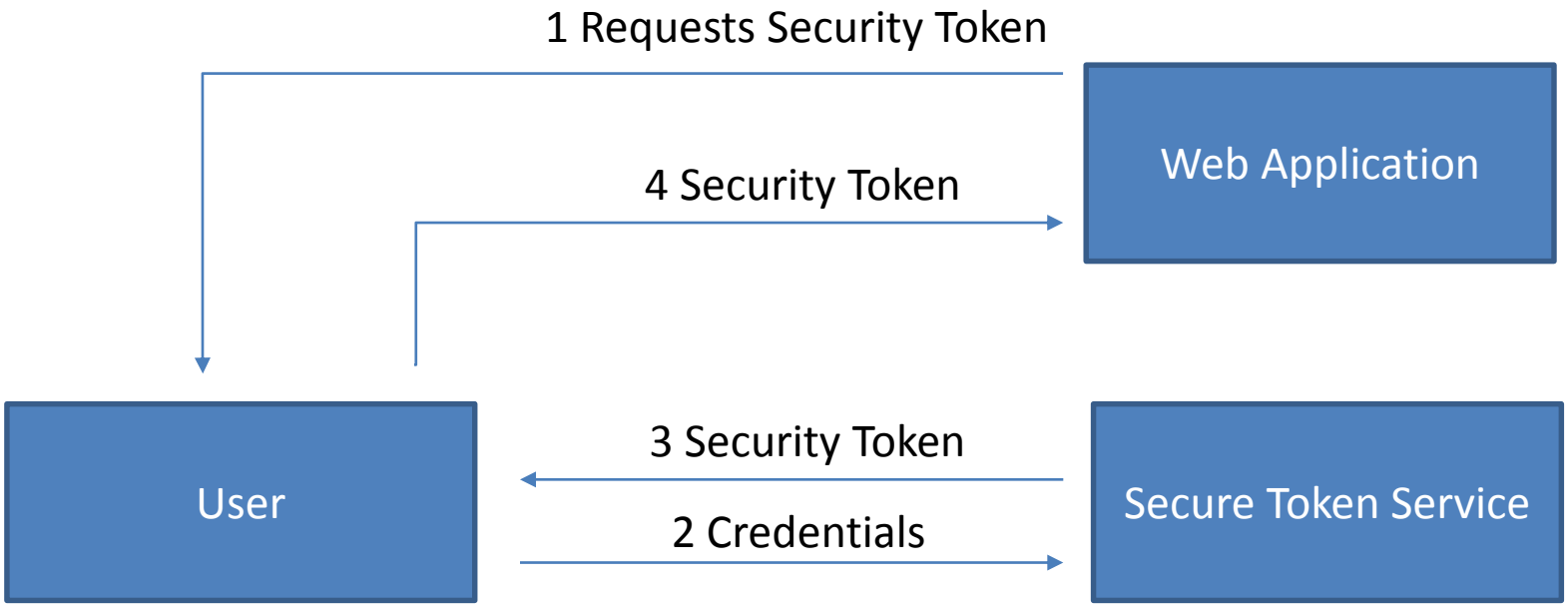
STS Flow illustration

If the credentials are valid, the STS will return a Security token



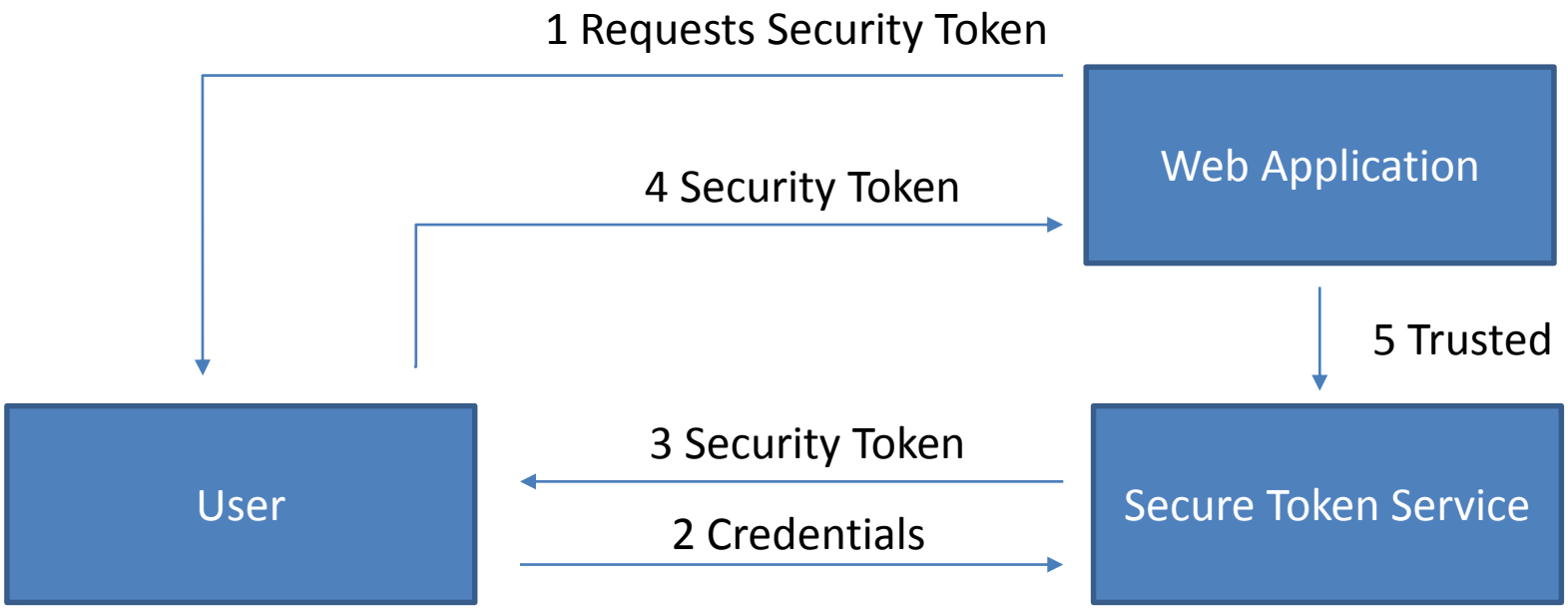
STS Flow illustration

The user provides the web application with the Security Token received from the STS



STS Flow illustration

The web application checks if the STS is a trusted issuer



STS Flow illustration

If the STS is trusted the web application returns an encrypted cookie to the user for authorization purposes