QR SOLVE

Cyber security

Use cases

END USER AUTHENTICATES TO GET ACCESS TO THE RESOURCES

End user with access granted authorises in the application using Strong Customer Authentication (SCA) / Two Factor Authentication (2FA) with two factors:

- something he knows (password / PIN)
- something he possesses (smartphone / hardware token)
- something he is (biometric: TouchID / Fingerprint / FaceID)

Authentication methods possible to use (combination):

- static password
- Personal Identification Number (PIN)
- digital certificate (x.509)
- mobile app
- RSA / Youbico hardware token
- Touch ID / Fingerprint
- Face ID
- Google / Microsoft 2FA Authenticator
- Apple ID
- Facebook
- One Time Passwords (OTP)/ SMS
- etc.

Use cases

THE DEVICE AUTHENTICATES TO GET ACCESS TO THE RESOURCES

The physical device as a part of IoT infrastructure authenticates to get access to the resources using simple authentication or 2FA

Authentication tools possible to use (combination):

- digital certificate (x.509)
- static password
- PIN number

THE ADMINISTRATOR AUTHENTICATES TO GET ACCESS TO THE RESOURCES

Backend user authenticates to get access to the resources placed in protected zone with simple (password/PIN) or with 2FA

THE ADMINISTRATOR MANAGES THE END USERS AND PRIVILEGES

The administrator manages the users inside the backoffice console or directly in authentication server console. He adds/remove/modify users, and defines their roles/privileges.

Use cases

- ## USER / DEVICE GET AUTHORISED ACCESS FOR SELECTED RESOURCES

  The authenticated user or the device get access to appropriate resources for his role / roles.
  When user or the device is authenticated, all privileges are calculated for him and embeded into token which is generated during authentication process.
  When user / device wants to get access to specific resources he always passes his token. The backend validates that token, check the permissions and:

  - allows to use the requested resources if he has permission and token is valid
  - block access to the requested resources if user has no permission or token is invalid / expired

Security

Data and infrastructure security

Data

- **Where data is stored** (Data at rest) incl. geographic locations which meets privacy regulations and privacy policies
- **Where data is exposed and consumed** (Data in transit). Active transmission of data across the network and locations
- **Data encryption** (when it's accessed and processed - data at rest/data in transit) with encryption keys
- **Tokenization** - Protection against exposing sensitive data. Instead of that hash/token is used.
- **Data isolation** (In multitenant model)

Infrastructure

- **Secured connections:** *Secure Socket Layer Protocol* (SSL), *Transport Layer Security* (TLS) as a standard and extension of SSL, *mutual TLS* (mTLS) as a extension of standard TLS. *Virtual Private Network* (VPN) as a secured connection between host and the server.
- **Restricted/public zones** *Virtual Private Cloud* (VPC) as isolated network, *demilitarised zone* (DMZ) - additional security layer for internal network protection, *Private Zone*
- **Network traffic:** *Subnets* - segmentation of VPC**,** *Access Control List* (ACL)
- **Shields and firewalls:** *AWS Shield* against DDoS attacks. *AWS WAF* monitoring and protection against web exploits and attacks
- **Hardware protection** hardware security module (*HSM*) enables you to easily generate and use your own encryption keys

Security

Application Backend and Frontend security

Backend

- **API security:** API can be open or secured. For secured API following security mechanisms support resources protection: (oAuth2, JWT, encryption, digital certificates)
- **Cross-services communication & async communication:** Low level communication based on queues and streamings (*Kafka, Kinesis, etc).* with restricted access to events
- **Access to the resources** *REST API HATEOAS level 3* means ONLY backend decides what actions are possible to do with specific resource, *Tokenisation* - tokens instead of real data are used, *RBAC* (Role Bases Access Control) principle
- **Component vulnerabilities:** State-of-the-art solutions and components without known and critical vulnerabilities.
- **Sensitive data exposure**

.

Frontend

- **Authorised access:** Authentication and authorisation before getting access to the resources, based on RBAC (role base access control)
- **REST API lvl 3:** Possible actions to do on specific resources are driven by the Backend
- **Compliance with OWASP:** Open Web Application Security Project publishes document outlining the 10 most critical security concerns for web application security.
- **Component vulnerabilities:** State-of-the-art solutions and components without known and critical vulnerabilities.
- **Web server configuration:** DDoS prevention, HTTP Methods whitelist, etc.
- **Tools:** Specialised tools for security issues identification like SonarQube/Cypress security scanners or BURP for attacks simulations

Compliance

Compliance with requlations and standards to provide the best security

Regulations

- Privacy policies
- Privacy regulations
- GDPR
- Acts
- PSD/PSD2
- etc.

Standards

- ISO 27001
- IEC 62443 - Industry
- IEC 61850 - Electric Power Utility
- RFCs
- etc.

Basics

Protection against unauthenticated and/or unauthorised access to IT resources.

Authentication

- **User authentication:** User identification (who uses the app)
- **User authentication tools,** allows to fulfil regulations and restrictions coming from privacy policies, i.e. static pass, PIN, OTP, digital certificates, mobile apps, hardware tokens, smartphones, etc.
- **Simple authentication**: good for the backend employees who works in private network
- **2 Factor Authentication** (2FA) / **Strong Customer Auth** (SCA): for authentication user uses at least two of the following auth tools categories: knowledge, possesion, inherence
- **Multi Factor Authentication (MFA)**: additional factors, like location and time
- **Machine authentication**

Authorization

- **Roles** the major privileges aggregation level. The most important in RBAC (Role Based Access Control) approach.
- **Privileges:** Determine possible actions to do by the user, aggregated into role.
- **Access control / RBAC** Access to the resources is restricted to specific role.
- **User rights management** Administrator can manage user rights (roles and privileges) directly in authorisation component/server or inside the bakcoffice dashboard console
- **Tracking activities:** Audit logs of user activities
- **API access control:** Authorised / Unauthorised

Integration

Integration with existing authorisation servers or legacy systems.

Single Sign-On

- **Centralised Auth Server:** One auth server for the organisation or all customers. Allows to keep one login credentials/ authentication tools / security standards for all users.
- **Redirection**: user is redirected to auth server login page and authenticates
- **Access to all systems (incl. legacy).**
- **Incremental migration**
- LDAP
- Open LDAP / Active Directory

Standards

State-of-the-art security standards

PKI / x.509

- Encryption / Digital signature
- Private & Public keys
- Digital certificates
- Access Control List
- Certificate Revocation List
- Qualified & self-signed certs

oAuth 2.0

- Auth flow for mobile, web, devices, desktop
- Access tokens
- Scope (grants)
- Protocols: OpenID Connect, SAML

IEC / ISO

- International Electrotechnical Commission
- Information Technology (IT)
- Operational Technology (OT)

OWASP

- Standard for Web Application
- OWASP top 10
- API Security
- Automated testing

Pen Tests

Investigation against software vulnerabilities

Components

- Web, mobile & API against OWASP top 10
- Advanced security testing against latest standards
- System integrations
- Infrastructure
- Data

Automation

- Continuously verification software vulnerabilities
- Static Code Analysis
- Quality Gates
- CI/CD integration

CEH

- Certified Ethical Hackers
- Code review
- Infrastructure review
- BURP (attacks simulation)*
- Red teaming*
- Black-Box, Grey-Box, White-Box*
- Reporting & recommendations*
-

* - available as additional service

Thank You

We're here to help you

Your proven Technology Partner

QR SOLVE

Address

Reformacka 6
35-026 Rzeszów
Poland

Telephone

+48 668 474 710
+48 694 150 492

Website

www.qrsolve.com