Securing Microservices with OpenID Connect and Spring Security

Workshop

Andreas Falk

Agenda

1. Concepts:
   - OAuth 2.0
   - OpenID Connect 1.0
2. Hands-On Part
   - Resource Server
   - Client (Authorization Code)
   - Client (Client Credentials)
3. Best Practices & Outlook
   - Security Best Practices
   - OAuth 2.1, OAuth 3.0

Workshop Source Code and Tutorial

Tutorial:

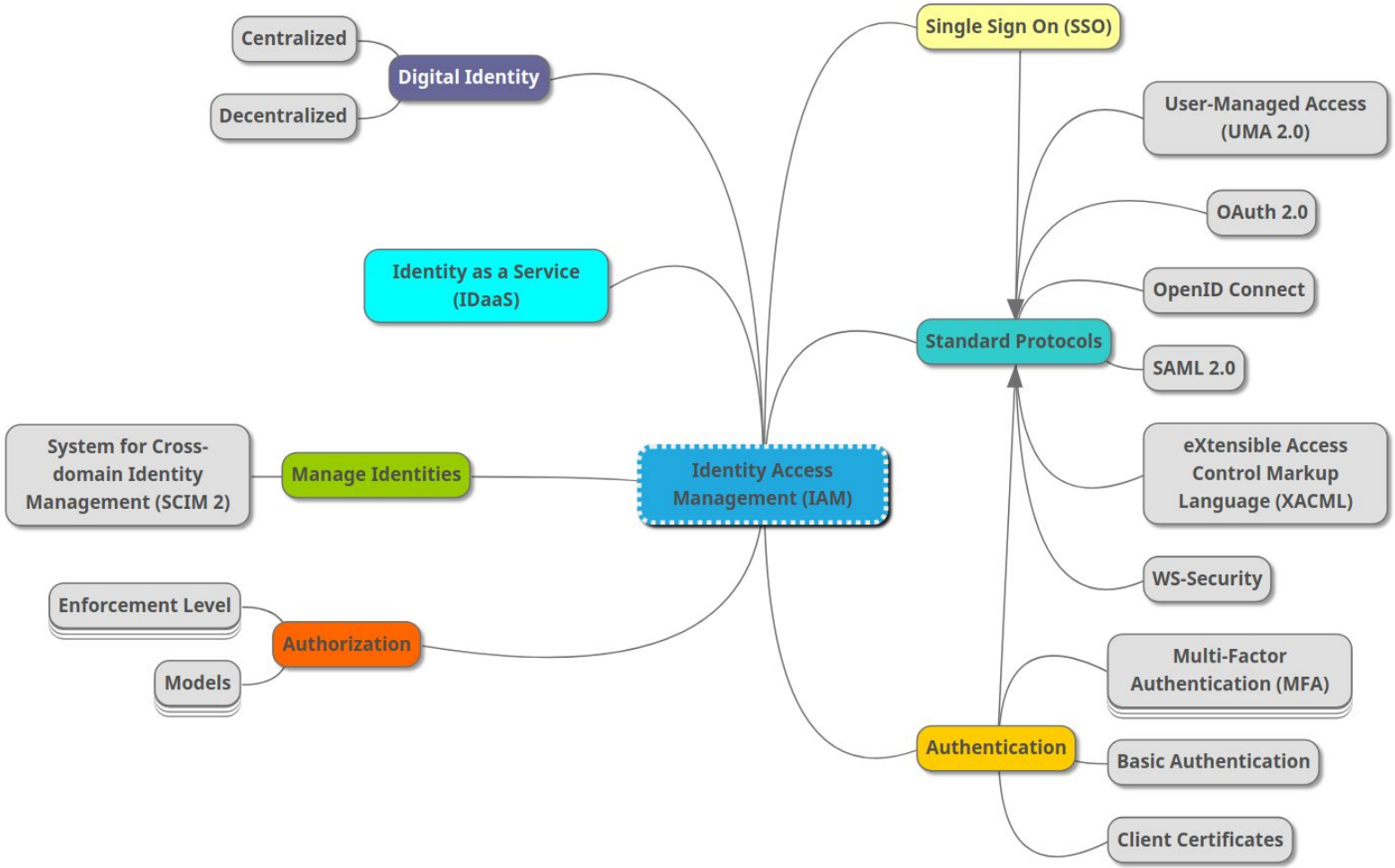
https://andifalk.gitbook.io/openid-connect-workshop

Source-Code:

https://github.com/andifalk/secure-oauth2-oidc-workshop

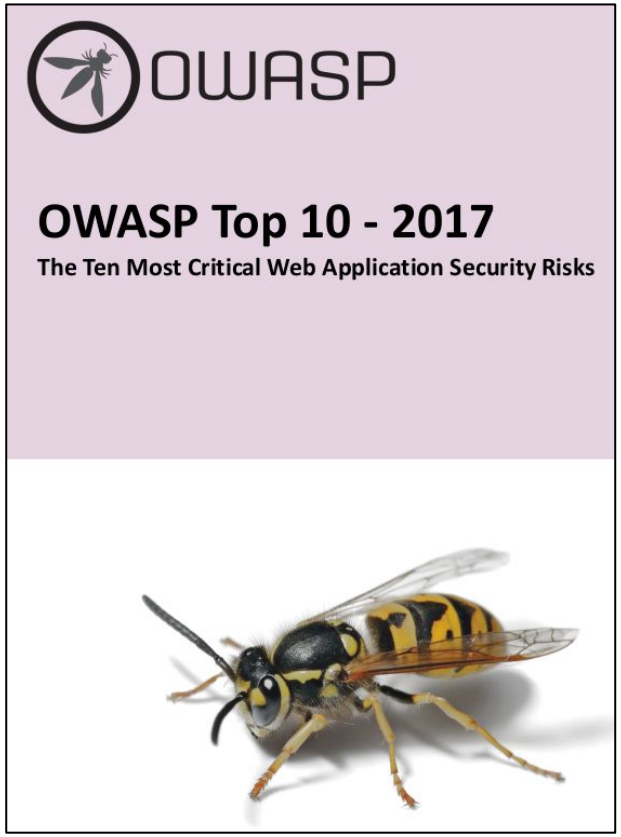
Authentication & Authorization

Introduction


Centralized
Digital Identity
Decentralized
Identity as a Service (IDaaS)
System for Cross-domain Identity Management (SCIM 2)
Manage Identities
Enforcement Level
Authorization
Models
Identity Access Management (IAM)
Single Sign On (SSO)
User-Managed Access (UMA 2.0)
OAuth 2.0
OpenID Connect
Standard Protocols
SAML 2.0
eXtensible Access Control Markup Language (XACML)
WS-Security
Multi-Factor Authentication (MFA)
Authentication
Basic Authentication
Client Certificates

NOVATEC

OWASP Top 10 2017



A2:2017-Broken Authentication

A5:2017-Broken Access Control

https://github.com/OWASP/Top10

NOVATEC

OWASP Application Verification Standard (ASVS)

V3.5 Token-based Session Management

Token-based session management includes JWT, OAuth, SAML, and API keys. Of these, API keys are known to be weak and should not be used in new code.

#	Description	L1	L2	L3	CWE	NIST §
3.5.1	Verify the application does not treat OAuth and refresh tokens — on their own — as the presence of the subscriber and allows users to terminate trust relationships with linked applications.		✓	✓	290	7.1.2
3.5.2	Verify the application uses session tokens rather than static API secrets and keys, except with legacy implementations.		✓	✓	798	
3.5.3	Verify that stateless session tokens use digital signatures, encryption, and other countermeasures to protect against tampering, enveloping, replay, null cipher, and key substitution attacks.		✓	✓	345	

https://github.com/OWASP/ASVS

NIST Special Publication 800-63B

Digital Identity Guidelines

Authentication and Lifecycle Management

5.1.1 Memorized Secrets

A Memorized Secret authenticator — commonly referred to as a *password* or, if numeric, a *PIN* — is a secret value intended to be chosen and memorized by the user. Memorized secrets need to be of sufficient complexity and secrecy that it would be impractical for an attacker to guess or otherwise discover the correct secret value. A memorized secret is *something you know*.

https://doi.org/10.6028/NIST.SP.800-63b
https://pages.nist.gov/800-63-FAQ

NOVATEC

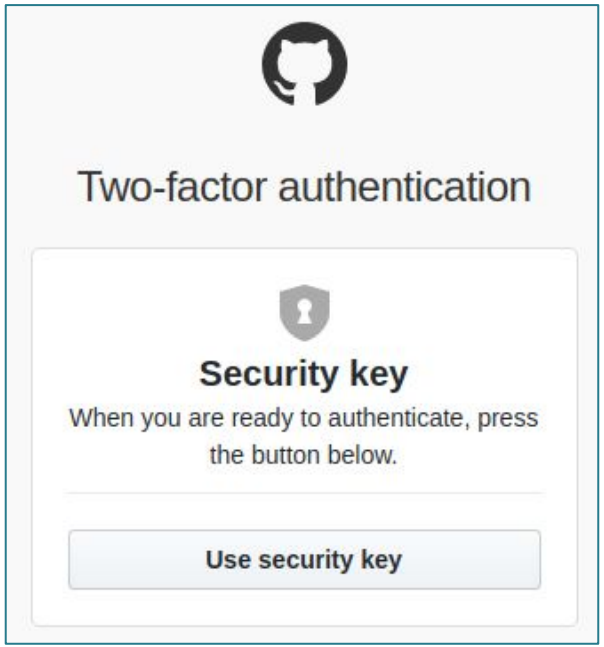
Authentication

Knowledge Factor (something the user knows):
Password, PIN, security question,...

Ownership Factor (something the user has):
ID card, security token, cell phone holding a software token,...

Inherence Factor (something the user is):
Fingerprint, retinal pattern,...

Multi-Factor Authentication



NOVATEC

Common Authentication Mechanisms

- Basic Authentication / Digest Access Authentication
- Form-based Authentication (i.e. using Session Cookies)
- Client-Certificates (Mutual TLS)
- Kerberos Tickets
- Proprietary mechanisms like API-Tokens, Siteminder etc.
- SAML Assertion Tokens
- JSON Web Tokens
- OAuth 2.0 & OpenID Connect 1.0
- WebAuthn / FIDO2

The Future of Identity (Identiverse Conference June 2020)

Past Predictions

1. The Need for Password Vaulting
2. SAML is Dead
3. The Year of PKI

Continuous Future

1. OIDC and SCIM are the New Normal
2. SAML will Still be Dead
3. Passwords will Also be Dead
4. WebAuthn will be an Alternative to Social Sign-On

Adjacent World

1. Active Clients
2. Quantum Computing & PKI
3. Balkanization of the Internet

Ian Glazer
VP Product Management - Salesforce
Founder/President - IDPro

NOVATEC

Thank You very much!
Questions?