Projektowanie obiektowe oprogramowania
Wykład 14 – Elementy architektury enterprise (1)
Single Sign-on
Wiktor Zychla 2012

---

1 Architektura aplikacji rozległych

Aplikacje rozległe (ang. *Enterprise applications*) – to wielomodułowe systemy informatyczne, często rozwijane przez lata lub powstające w wyniku połączenia kilku niezależnych elementów, zbudowanych w różnych technologiach i w oparciu o różne konstrukcje architektury.

Najprostszy przykład – połączenie systemów informatycznych dwóch (lub więcej) dużych banków. Inny przykład – zintegrowany miejski/gminny/powiatowy system informatyczny, obejmujący różne obszary odpowiedzialności podmiotu Zamawiającego.

W obszarze architektury systemy rozległe rodzą wyzwania **integracyjne**. Integracja z kolei oznacza przepływ informacji wewnątrz systemu – np. przepływ danych między modułami składowymi lub przepływ informacji o tożsamości użytkownika.

2 Single sign-on

Single sign-on (pojedyncze logowanie) – to właściwość aplikacji rozległych, w których dostęp do tych części poszczególnych modułów które wymagają autentykacji i autoryzacji, możliwy jest po jednokrotnym potwierdzeniu tożsamości użytkownika.

Z uwagi na różne implementacje realizujące ten sam cel, można mówić o wzorcu dla aplikacji rozległych.

Najprostsza, na co dzień spotykana implementacja SSO wbudowana jest w systemy operacyjne – po jednokrotnym zalogowaniu dostaje się dostęp do aplikacji, które o tożsamość użytkownika odpytują system operacyjny. Takie SSO nie jest interesujące, ciekawie robi się dopiero wtedy, kiedy mówimy o SSO poza granicami jednego systemu – na przykład kiedy interfejs użytkownika osadzony jest w przeglądarce internetowej i dostaje się on do różnych witryn, rozsianych gdzieś po świecie.

Istnieją różne możliwości implementacji tego wzorca. Jednym z ważniejszych kryteriów właściwego wyboru jest zgodność ze standardami przemysłowymi.

Wśród powszechnie akceptowanych protokołów SSO należy wymienić:

- OpenID – dobry wybór ale phising + brak single sign off
- OAuth
- CAS (Central Authentication Service)
- SAML-p

- Shibboleth
- **WS-Federation –** Office365, Sharepoint 2010, Windows 8, Azure Cloud Services

3 Claims-based authentication

Claim (stwierdzenie/oświadczenie) – informacja o **Kimś** wydane przez jakiegoś **Wystawcę.**
Claim powinien być „podpisany" tzn. nie powinno być wątpliwości że wydał go **Wystawca.**

Zwykle nie da się nijak inaczej stwierdzić czy claim jest prawdziwy czy nie, ale **ufamy** wystawy wobec czego **akceptujemy** informację.

Przykład: Stwierdzenie – „*Jan Kowalski urodził się 04.11.1978*".

Jest to oświadczenie z „podpisem", powszechnie akceptowane w bankach, sklepach itd. Fakt akceptowania wynika z relacji zaufania do Wystawcy oświadczenia.

4 WS-Federation

4.1 Pojęcia

Protokół WS-Federation przenosi pojęcia „oświadczenia" i „wystawcy" na język techniczny:

Security Token Service (**STS**) – wystawca oświadczeń , posiada informacje o użytkownikach aplikacji rozległej lub zna lokalizacje innych wystawców

Oświadczenie – czwórka (Type, Issuer, Subject, Value)

Security Assertion Markup Language (**SAML)** – dialekt XML zapisu oświadczeń, standaryzujący m.in. ich podpisywanie cyfrowe (X509). SAML mówi tylko tym jak skonstruowane są tokeny. Nie mówi o tym jak je wymieniać (język vs protokół). Na SAML opartych jest kilka różnych protokołów: WS-Federation, Google SSO, Shibboleth, SAML-protocol)

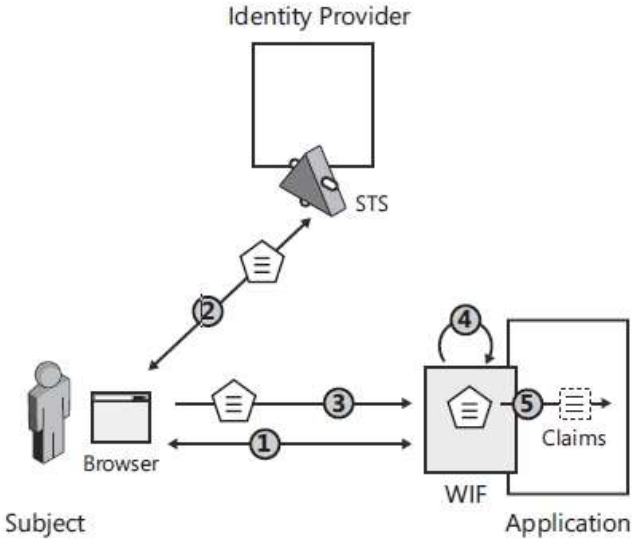
Typowe oświadczenia – nazwa użytkownika, imię, nazwisko, e-mail, adres, role (uprawnienia)

Security token (token bezpieczeństwa) – zbiór oświadczeń

Relying Party (**RP**)– aplikacja która ufa claimom wydanym przez STS

Mówi się „RP jest sfederowany (*federated*) z STS" = RP ufa oświadczeniom wydanym przez STS. W praktyce jest to równoważne stwierdzeniu „RP akceptuje zbiór czwórek (Type, Issuer, Subject, Value) podpisany znanym mu certyfikatem STSa".

4.2 Protokół pojedynczego logowania



1. Użytkownik próbuje uzyskać dostęp do części systemu rozległego wymagającej autoryzacji. Aplikacja wymusza przekierowanie sesji przeglądarki do aplikacji – wystawcy oświadczeń
2. Wystawca oświadczeń weryfikuje tożsamość użytkownika (lub wykorzystuje fakt że tożsamość była już sprawdzana wcześniej), tworzy podpisany token SAML i przekazuje go do przeglądarki
3. Przeglądarka przekazuje otrzymany token do aplikacji wymagającej autoryzacji
4. Aplikacja wykorzystuje technologię umożliwiającą przetwarzanie oświadczeń (tu: Windows Identity Foundation) (lub przetwarza oświadczenia samodzielnie) w tym weryfikuje poprawność ich podpisu
5. Zestaw oświadczeń jest dostępny dla aplikacji

4.3 Protokół pojedynczego wylogowywania

1. Wystawca oświadczeń śledzi żądania wydania tokenu bezpieczeństwa – magazynuje adresy aplikacji występujących o oświadczenia
2. Po otrzymaniu żądania *wylogowania*, wystawca generuje do przeglądarki zasób (stronę) zawierającą adresy wszystkich aplikacji, które dotychczas w imieniu użytkownika wystąpiły o token bezpieczeństwa, ale dodaje do tych adresów parametr oznaczający wylogowanie (tu: *wsignoutcleanup1.0*).
3. Przeglądarka kieruje żądania do wszystkich kolejnych aplikacji

4. Aplikacje wykonują sobie tylko znaną procedurę wylogowania użytkownika z sesji

4.4 Bezpieczeństwo protokołu

Bezpieczeństwo protokołu WS-Federation oparte jest o 4 certyfikaty X509 (wszystkie poza jednym są opcjonalne):

- (O) Certyfikat bezpiecznych połączeń do serwera aplikacji (SSL)
- (O) Certyfikat bezpiecznych połączeń do serwera wystawcy oświadczeń (SSL)
- Podpisanie oświadczeń przez wystawcę oświadczeń (podpisany SAML)
- (O) Szyfrowanie wystawianych oświadczeń certyfikatem aplikacji

4.5 Inne cechy protokołu

Relacja zaufania do wystawcy jest przechodnia – jeżeli klient (RP) prosi o oświadczenia, a wystawca (STS) przekieruje jego żądanie do kolejnego wystawcy (a ten z kolei dalej itd.) to w efekcie ostateczny zbiór oświadczeń może być sumą oświadczeń wydanych przez kolejnych wystawców, a klient w ogóle nie musi być świadomy tego przez ile „węzłów" wystawców przeszło żądanie.

To daje możliwość budowania „bramek" (gateway), które na zewnątrz (dla klienta) implementują protokołów WS-Federation, a wewnątrz pozyskują oświadczenia albo od innego wystawcy WS-Federation albo z usługi logowania innego protokołu.

5 Demos

1. Budowa prostej aplikacji wystawcy oświadczeń – własny STS oparty o Forms Authentication
2. Metadane aplikacji wystawcy oświadczeń
3. Federacja aplikacji opartej o logowanie zintegrowane – fedutil.exe (Add STS Reference…)
4. Federacja aplikacji opartej o logowanie typu Forms
5. Single sign-out
6. Oświadczenia lokalne – **CustomClaimsAuthenticationProvider**
7. Omówienie przykładowych implementacji przemysłowych I możliwych scenariuszy:
    a. Active Directory Federation Services 2, panel konfiguracji
    b. Thinktecture IdentityServer

6 Literatura

Patterns & Practices – „A Guide to Claims-based Identity and Access Control" (darmowy ebook), http://msdn.microsoft.com/en-us/library/ff423674.aspx

Vito Bertocci – "Programming Windows Identity Foundation"