Federated Identity met Azure Access Control

Patriek van Dorp

De applicaties die we vandaag de dag bouwen zijn gebaseerd op internet technologie. Iedere applicatie bevat zijn eigen logica om de identiteit van gebruikers te valideren en om gebruikers te autoriseren voor bepaalde functionaliteit van de applicatie. Hoe externaliseren we de logica voor het authenticeren en het autoriseren van gebruikers, zodat we ons meer kunnen richten op de behoeften van de business?

Door de jaren heen hebben we vele complexe aspecten van de communicatie tussen hardware en software weten te abstraheren. Zo hebben we nu object geörieënteerde programmeertalen, zodat we niet meer hoeven te werken met lange reeksen enen en nullen. Nog een voorbeeld zijn device drivers die ervoor zorgen dat we niet meer al onze applicaties hoeven aan te passen als we een nieuw apparaat in onze computer stoppen.

Authenticatie is echter altijd achtergebleven. Iedere applicatie die we momenteel schrijven heeft immers weet van een gebruikersdatabase en weet hoe de eigenschappen van een gebruiker, de attributen, kunnen worden achterhaald. Dit is logica die helemaal niets te maken heeft met de functie die de applicatie voor een bedrijf heeft. Kortom, de kosten voor het implementeren van een authenticatiemechanisme kunnen niet worden gekoppeld aan inkomsten. Bovendien moet een gebruiker nu voor meerdere applicaties verschillende gebruikersnamen en wachtwoorden onthouden.

Claims-based Identity

In ons dagelijks leven wordt onze identiteit heel anders vastgesteld dan in de traditionele authenticatiemethodes die we gebruiken om de identiteit van de gebruikers van onze applicaties vast te stellen. In onze applicaties hebben we vaak te maken met gebruikers met rollen die in groepen zitten, die weer in groepen zitten, enzovoorts.

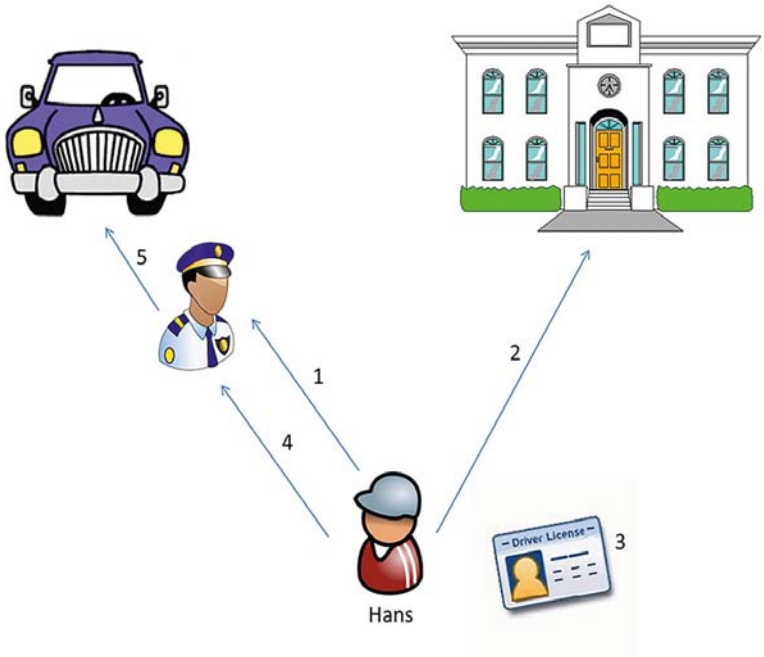
In het dagelijks leven is onze identiteit vaak veel complexer. We moeten bijvoorbeeld ouder zijn dan 21, langer zijn dan 1,20 meter of Japans kunnen spreken. Dit soort beweringen over een onderwerp (een gebruiker, een device of een service) noemen we claims.

> “We moeten in het dagelijks leven bijvoorbeeld ouder zijn dan 21, langer zijn dan 1,20 meter of Japans kunnen spreken”

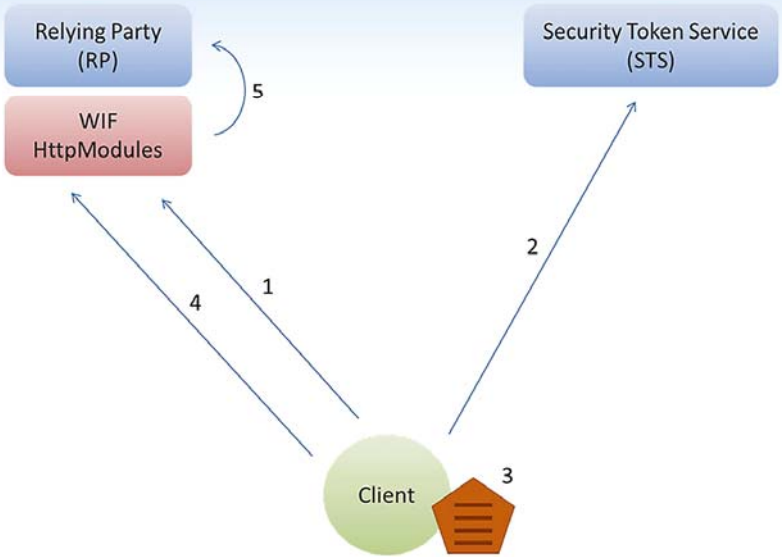
Een claim kan dus nog steeds iets zeggen over de rol die iemand binnen een organisatie speelt, maar claims kunnen ook complexer zijn. Het antwoord op de vraag ‘Is het onderwerp ouder dan 21?’ onthult bijvoorbeeld de geboortedatum van het onderwerp niet.

WS-Federation

Claims worden uitgegeven door een een instantie die vertrouwd wordt de juiste identiteit van een onderwerp te achterhalen. Laat mij dit duidelijk maken aan de hand van een voorbeeld uit het dagelijks leven.
In figuur 1 wil Hans een stukje gaan rijden in zijn nieuwe auto. Hij moet hiervoor wel eerst in staat zijn een rijbewijs te laten zien aan de politieman (1). Hans heeft wel zijn rijexamen gehaald, maar hij heeft zijn rijbewijs nog niet opgehaald. Hans moet dus eerst naar het gemeentehuis (2) om zijn rijbewijs op te halen. Het gemeentehuis (de issuer) valideert de identiteit van Hans door Hans te vergelijken met een recente foto in een eigen databestand



FIGUUR 1: UITGIFTE VAN EEN RIJBEWIJS.



FIGUUR 2: APPLICATIE AUTHENTICATIE.

en nog een paar andere criteria. Als de identiteit van Hans is geverifieerd, geeft het gemeentehuis het rijbewijs uit (3). Dit rijbewijs bevat een recente foto van Hans, zijn geboortedatum en de datum waarop het rijbewijs verloopt.

Nu kan Hans zijn rijbewijs overleggen aan de politieman (4). De politieman controleert of het rijbewijs van Hans is door de foto te vergelijken. Omdat het rijbewijs een groot aantal echtheidskenmerken heeft die moeilijk te vervalsen zijn en omdat het rijbewijs is afgegeven door een geloofwaardige instantie (het gemeentehuis), accepteert de politieman het rijbewijs. Hij controleert ook nog of het rijbewijs niet is verlopen en of het rijbewijs geldig is voor het type vervoersmiddel dat Hans wil gaan besturen (de auto). Als dit allemaal in orde is, mag Hans in zijn auto stappen om een stukje te gaan rijden (5).

Als we het voorbeeld van Hans nu converteren naar een voorbeeld van authenticatie in onze applicaties, ontstaat het volgende plaatje.

In figuur 2 wil de client (Hans) toegang tot onze applicatie (de auto) die ook wel relying party (RP) wordt genoemd (1). Doordat in de configuratie van onze applicatie een aantal Windows Identity Foundation (WIF) HttpModules (de politieman) zijn geconfigureerd die voor ieder request controleren of de client (ofwel het subject) geauthenticeerd is, is de applicatie vrij van authenticatielogica en kan de ontwikkelaar zich helemaal richten op de business requirements.

Als een subject niet geauthenticeerd is wordt de client geredirect naar een geconfigureerde Security Token Service (het gemeentehuis) of STS (2). De STS verifieert de identiteit van de gebruiker met een eigen gebruikersdatabase. Dit kan door middel van gebruikersnaam/wachtoord, maar de STS is vrij om hier ieder denkbare authenticatiemethode voor te gebruiken, inclusief het redirecten van de gebruiker naar een andere STS. Als de identiteit van de gebruiker is vastgesteld kunnen de vereiste attributen (claims) van de gebruiker in een SecurityToken worden gezet (3). Een SecurityToken wordt digitaal gesigneerd, waardoor het ongeldig zal zijn als het onderweg wordt aangepast (de echtheidskenmerken van een rijbewijs). Vervolgens wordt de client geredirect naar onze applicatie, maar deze keer bevat het request de SecurityToken die is verkregen van de STS (4). Omdat deze STS in onze applicatieconfiguratie bekend is als een betrouwbare Identity Provider (IP) en het SecurityToken aan alle eisen voldoet, wordt de client toegelaten tot onze applicatie (5).

In de voorgaande voorbeelden heb ik een paar stappen uit het WS-Federation protocol overgeslagen, maar de belangrijkste stappen staan erin.

1. Request een .aspx pagina in de applicatie;
2. Redirect naar de geconfigureerde STS en verschaf inloggegevens;
3. Ontvang een SecurityToken van de STS;
4. Redirect naar de .aspx pagina met het SecurityToken in de request;
5. Valideer het SecurityToken in de request.

Single Sign-On

Bij het inloggen op de STS wordt er een cookie weggeschreven op de client. Als we nu een andere applicatie proberen te benaderen die ook geconfigureerd is om te redirecten naar dezelfde STS, zal de STS zien dat we al geauthenticeerd zijn, omdat de cookie wordt gevonden. De STS zal meteen doorgaan naar het genereren van een SecurityToken, zonder dat de gebruiker opnieuw hoeft in te loggen. Door dit principe wordt Single Sign-On bereikt.

Identity Provider of Federation Provider

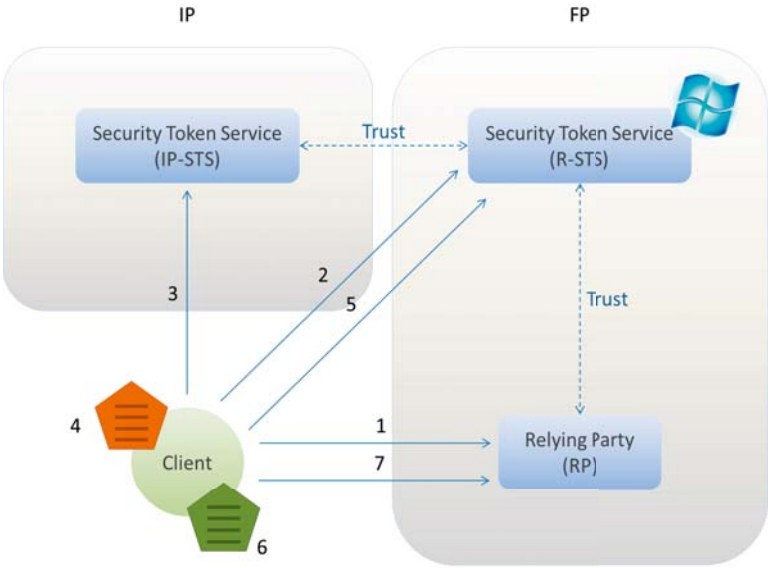
Er zijn twee verschillende soorten Security Token Services, Identity Providers en Federation Providers. Identity Providers (IPs)

> Identity Providers stellen de identiteit van een subject vast door bijvoorbeeld gebruikersnaam en wachtwoord te valideren tegen een lokale database.

stellen de identiteit van een subject vast. Dit gebeurt bijvoorbeeld door een gebruikersnaam en wachtwoord te valideren tegen een lokale database. IPs kunnen natuurlijk ook gebruikers attributen teruggeven.
Er is nog steeds een probleem als er meerdere applicaties gebruik maken van dezelfde STS of als er een STS wordt toegevoegd of verwijderd. In dit geval moeten alle applicatie alsnog één voor één worden aangepast en opnieuw worden geconfigureerd. Figuur 3 geeft schematisch weer hoe we dit kunnen oplossen.

In figuur 3 wil de client toegang tot een RP (1). De WIF configuratie van de RP (onze applicatie) zorgt ervoor dat de client wordt geredirect naar een zogenaamde R-STS (2). De R-STS kan aan de hand van het request van de client bepalen naar welke IP-STS de client moet worden geredirect en doet dit vervolgens (3). De gebruiker moet zich identificeren bij de IP-STS en als de gebrui-

ker is ge-authenticeerd genereert de IP-STS een SecurityToken (4). Omdat er geen vertrouwensrelatie is tussen de RP en de IP-STS zou de RP het SecurityToken van de IP-STS afwijzen. Dit komt overeen met het afhalen van je rijbewijs bij de slager. Dit zou niemand accepteren als een geldig identificatiebewijs. Daarom wordt de SecurityToken eerst verstuurd naar de R-STS (5). De R-STS kan logica bevatten om de claims uit de SecurityToken te converteren als dit nodig is. Als de RP bijvoorbeeld vereist dat een gebruiker een 'role' claim heeft met de waarde 'Administrator' en de IP-STS staat in China, dan zullen de chinese karakters voor 'Administrator' eerst moeten worden vertaald. Dit zou logica kunnen zijn die een R-STS bevat. Vervolgens verpakt de R-STS alle geconverteerde claims in een nieuwe SecurtyToken (6). Dit SecurtyToken wordt wel vertrouwd door de RP en kan dus worden gevalideerd (7).



FIGUUR 3: FEDERATED IDENTITY.

Azure AppFabric Access Control Service

Windows Azure AppFabric Access Control Service (ACS) is een voorbeeld van een in hoge mate beschikbare en uiterst schaalbare R-STS. In ACS kunnen meerdere IP-STSs geconfigureerd worden. Gebruikers kunnen dan kiezen welke IP-STS ze willen gebruiken om in te loggen. Claims, afkomstig van de IP-STS kunnen worden vertaald of geconverteerd. Er kunnen meerdere RPs geconfigureerd worden die individueel gekoppeld kunnen worden aan IP-STSs. Op deze manier worden er dus tweevoudige trustrelaties opgezet tussen RP en IP-STS door enkel aanpassingen in de R-STS te doen. IP-STS noch RP hoeven aangepast te worden. Alle applicaties in het applicatieportfolio van een bedrijf verwijzen naar ACS en ACS zorgt ervoor dat de juiste claims bij de juiste applicaties komen.

Patriek van Dorp, is senior technology specialist op het gebied van Microsoft technologieën bij Sogeti Nederland. Mail: patriek.van.dorp@sogeti.nl. Twitter: @pvandorp.

nieuws

Nokia geeft nieuwe impuls aan WP7

Nokia en Microsoft starten met de vorige maand aangekondigde samenwerking een offensief op de mobiele markt. Zij zullen de kennis op hard- en softwaregebied bundelen en zodoende nieuwe producten en diensten kunnen bieden. Beide concerns denken al op korte termijn de vruchten van de samenwerking te kunnen plukken, zo blijkt uit uitlatingen van CEO Steven Ballmer van Microsoft en van Nokia-CEO Stephen Elop.
De overeenkomst zal belangrijke gevolgen hebben:

- Nokia Windows Phone adopteert als haar belangrijkste strategie voor de smartphone.
- Windows Phone krijgt een flinke stimulans van expertise, hardware-design, en taalondersteuning, die door Nokia wordt ingebracht. Ook het aantal verkooppunten zal sterk toenemen.
- Bing wordt de standaard zoekmachine op de Nokia toestellen
- Microsoft adCenter gaat de advertising op de Nokia's verzorgen.
- Nokia Maps wordt het geografische hart van de toestellen.
- De overeenkomsten van Nokia met operators op het gebied van betaaltransacties maken het voor klanten makkelijk in landen waar het gebruik van credit cards nog laag is.
- De Microsoft development tools zullen worden gebruikt om nieuwe applicaties voor de Nokia smartphones te schrijven.

Visual Studio 2010 SP1

Sinds 10 maart is Service Pack 1 van Visual Studio 2010 officieel beschikbaar. Service Pack 1 bevat een grote hoeveelheid fixes voor issues die via Microsoft Connect zijn aangemeld en waarop gebruikers van Visual Studio konden stemmen.
In deze release is inbegrepen een verbeterde Help ondersteuning, ondersteuning van IntelliTrace voor 64bit en SharePoint en de Silverlight 4 Tools. Daarnaast is ook ondersteuning voor unit testing op .NET 3.5 beschikbaar en een nieuwe performance wizard for Silverlight. Tevens zijn er een tweetal feature packs beschikbaar gekomen voor MSDN Subscribers namelijk het Team Foundation Server Project Server Integration Feature Pack en Visual Studio 2010 Load Test Feature Pack;
Het Team Foundation Server Project Server Integration Feature Pack biedt een integratie tussen Microsoft Project Server and Team Foundation Server voor een verbeterde samenwerking tussen het Project Management Office en het Project team .