Einfluss von externer Autorisierung auf die Performanz in OAuth2 Systemen

Mit OAuth2 ist eine Sicherung von http-Schnittstellen möglich. Hierbei erhält ein Client von einem Autorisationsserver einen Token, mit dem er Zugriff auf Schnittstellen eines Resourceservers erhält. Ein valider Token ist mit einer Authentifizierung gleichzusetzen. Die Autorisierung, also die Entscheidung, ob der Token die benötige Berechtigungen besitzt, lässt sich grundsätzlich in dem Resourceserver selbst implementieren. Dies kann aber bei einer heterogenen Applikationslandschaft mit verschiedenen Programmiersprachen und sich häufig wechselnden und komplexen Zugriffsrichtlinien schnell zu einem hohen Wartungsaufwand führen. Deswegen ist es sinnvoll die Autorisierung zu entkoppeln und dies ist mit „Open Policy Agent“ (OPA) möglich. Das hat allerdings zur Folge, dass der Resourceserver jedes Mal bei eingehenden http-Anfragen den OPA-Service den u.U. umfangreichen Token zusenden muss, der wiederum dann den Token dekodieren und parsen und eine Zugriffsentscheidung zurücksenden muss, d.h. es besteht das Risiko von Performanceeinbußen. Da bei einer hohen Last auf Schnittstellen diese i.d.R. durch horizontale Skalierung entlastet werden, können Performanceeinbußen äußerst kostspielig sein.

OPA gibt an, dass basierend auf Benchmarks Evaluierungen von Zugriffsentscheidungen selbst nur lediglich Rechenzeit im Bereich von einer Millisekunde benötigen (Agent, 2021). Allerdings wird hier nicht die in der Praxis relevante Latenz bzw. Response Time berücksichtigt, nämlich die Zeit, die benötigt wird, wenn ein Client eine http-Anfrage an den Ressourceserver sendet, dieser dann die Anfrage, um eine Zugriffsentscheidung zu erhalten an den OPA-Service sendet, um dann schlussendlich dem Client eine Antwort auf seine Anfrage zu senden. Um den Einfluss von externer Autorisierung im Vergleich zur applikationsinternen Autorisierung zu untersuchen, wurden zwei Testsysteme implementiert. In dem einen wird die Autorisierung in der Applikation gehandhabt und in dem anderen entkoppelt durch OPA. Um Last-Stress-sowie-Skalierbarkeitstests durchzuführen, wurde Apache JMeter verwendet und hierbei insbesondere die Latenz sowie Response Time als zu betrachtende Metrik gewählt.

Dabei ist man zu dem Ergebnis gekommen, dass …

Literaturverzeichnis

Agent, O. P. (6. Juli 2021). *openpolicyagent*. Von openpolicyagent: https://www.openpolicyagent.org/docs/latest/policy-performance/ abgerufen