

1. Task

Meine Aufgabenstellung lautete folgendermaßen: In dCache 2.6 wurde zu einer neuen Version der Berkeley-Datenbank gewechselt. Da diese mit der Vorgängerversion binärinkompatibel war, wurde ein Migrationsskript hinzugefügt. Das Skript ebenso wie das alte DB-Jar sind in dCache Version 2.7 überflüssig und sollen gelöscht werden.

Da diese Aufgabe überschrieben mit dem Stichwort "git based archaeology" überschrieben war und ich bis zu diesem Zeitpunkt nur oberflächliche Erfahrung mit dem git-Plugin in Eclipse hatte, habe ich mich zunächst mit git über die Kommandozeile vertraut gemacht und mir viel Zeit genommen verschiedene Tutorials durchzuarbeiten.

Mittels git log habe ich mir die zu den genannten Dateien gehörenden commits angeschaut und Referenzen auf die Dateien im sonstigen Code geprüft. Da die hierzu gehörige Exception mit Verweis auf die neue Datenbank und die notwendige Migration bereits von einem Entwickler gelöscht worden war, blieben nur die zwei Dateien zu löschen. Anschließend habe ich einen Pull Request für meinen passend benannten Entwicklungsbranch gestellt, der nach Klärung einer Rückfrage auch gemergt wurde.

Ich habe diese eher kleine Aufgabe zum Anlasse genommen, mich wirklich gründlich in git einzuarbeiten und mich anhand der genannten Dateien in einem Teil des dCache-Codes genauer umzusehen und die Zusammenhänge zwischen den Klassen und deren Veränderung über die Zeit nachzuvollziehen.

2. Vortrag: Thema Macaroons

Macaroons wurden 2014 von einem Entwicklerteam von Google als "Cookies with Contextual Caveats for Decentralized Authorization in the Cloud" vorgestellt: Wie Cookies sollen sie zur Autorisierung dienen, nach einem recht simplen Mechanismus funktionieren und für Entwickler leicht nutzbar sein. Darüber hinaus sind sie aber speziell für den Einsatz in verteilten Systemen geeignet, d.h. wie hier beschrieben für die Cloud, aber auch allgemein für mobile und Webanwendungen, bei denen eine lose Kopplung angestrebt wird. Zusätzlich bieten sie den Vorteil, dass die Bedingungen für eine Autorisierung sehr präzise bestimmt werden können, indem dem Macaroon eine beliebige Menge zu erfüllender Bedingungen angefügt werden. Diese können beim 'Weiterreichen' des Macaroons (Delegation) erweitert, nicht aber verringert werden.

Einschränkungen der Gültigkeit eines Macaroons können sich auf den Erstellungszeitpunkt des Macaroons beziehen (z.B. nicht älter als 10 Sekunden), es kann eine bestimmte IP-Adresse gefordert werden, nur Lesezugriff (kein Löschen) gewährt werden oder nur genau eine Datei aus einer Sammlung freigegeben werden. Die Gültigkeit aller gegebenen Voraussetzungen wird vom Ersteller des Macaroons geprüft, sobald dieser dessen formale Korrektheit auf Basis einer kryptografischen Hash-Funktion festgestellt hat.

Neben diesen First-Party-Caveats gibt es auch Third-Party-Caveats, d.h. Bedingungen, die nicht der Ersteller selbst prüft, sondern deren Prüfung er Drittparteien anvertraut. So kann eine Anwendung anstatt eines eigenen Vorgangs zum Beispiel auf die externe Authentifizierung via Facebook- oder Hochschulaccounts setzen. Der Drittanbieter bestätigt dem Client seine Identität mittels eines Dispatch-Macaroons, dass dieser zusätzlich zu dem vom eigentlichen Servicenbieter erstellten Macaroon bei diesem wieder "einreicht".

Fazit: Macaroons sind leichtgewichtig und schnell und können per Request erzeugt werden. Durch die präzise Beschränkung ihrer Gültigkeit sind sie flexibel einsetzbar und sicher. Die Möglichkeit zur Einbindung von Drittanbietern ermöglicht die angestrebte lose Kopplung verteilter Systeme.

Quellen:

Birgisson, Arnar et al. (2014): „Macaroons: Cookies with Centextual Caveats for Decentralized Authorization in the Cloud.

<https://static.googleusercontent.com/media/research.google.com/de//pubs/archive/41892.pdf>

Erlingsson, Ulfar (2014): Macaroons: Cookies with Centextual Caveats for Decentralized Authorization in the Cloud. Talk at „Air Mozilla“ <https://air.mozilla.org/macaroons-cookies-with-contextual-caveats-for-decentralized-authorization-in-the-cloud/>

Nacionales, Jonathan (2015): Scalable Cross-Domain “Cookie” Authentication/Authorization: An Example with macaroon.js
<https://iamalivingcontradiction.wordpress.com/2015/11/09/scalable-cross-domain-cookie-authenticationauthorization-an-example-with-macaroon-js/>