Durch eine Blockchain und ein Peer-to-Peer-System gesicherte Online-Ausleihe

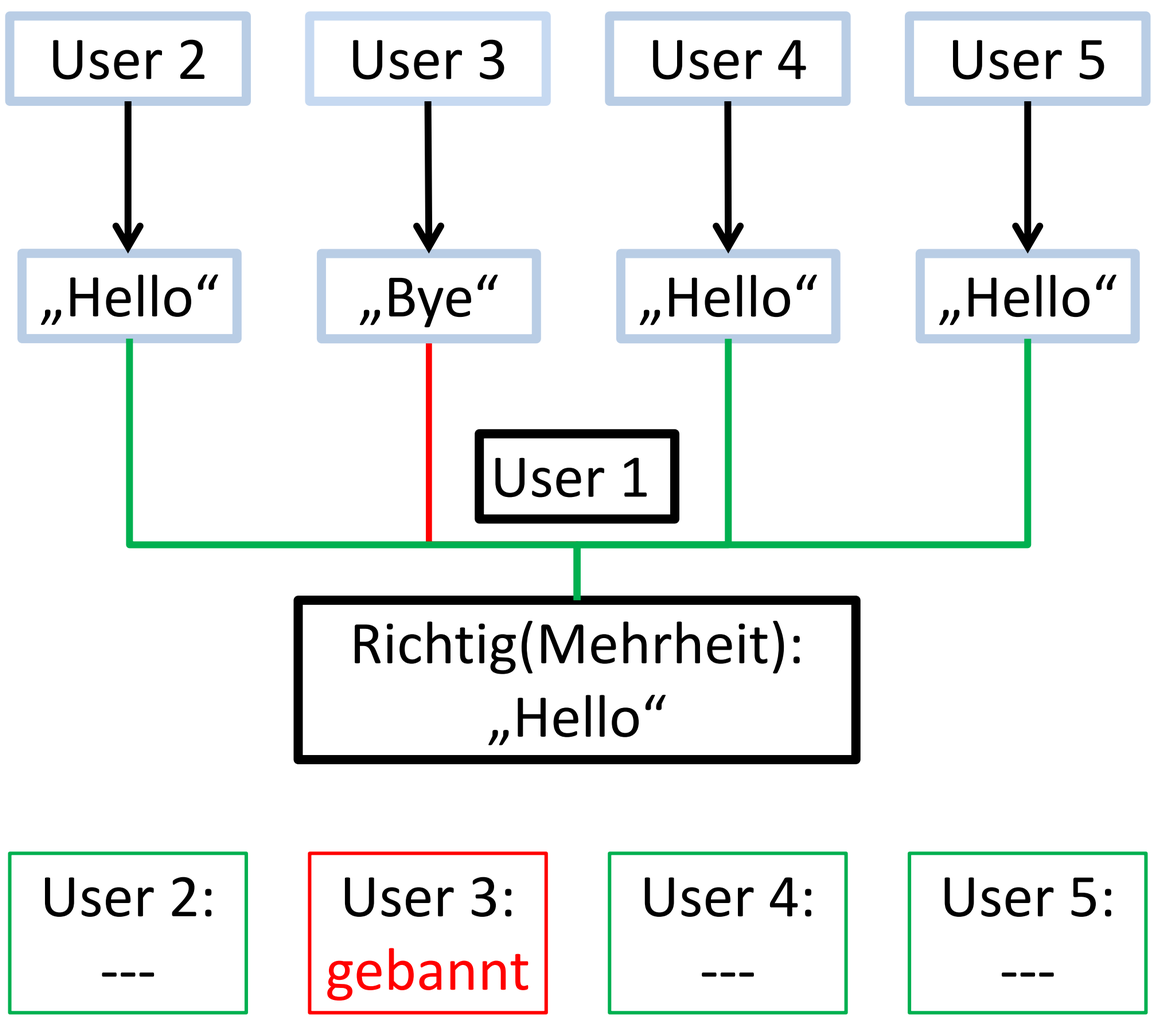
Alexander Reimer (13) und Matteo Friedrich (12)
Gymnasium Eversten Oldenburg

Forscherziel

Wir haben versucht, eine Blockchain und ein Peer-to-Peer-Netzwerk zu programmieren, um damit eine sichere Online-Ausleihe für unsere Schulbibliothek zu erschaffen. Das sind neuartige Techniken, mit denen man Daten sicher zwischen verschiedenen Computern synchronisieren kann. Mit sicher ist gemeint, dass diese Daten „unhackbar“, also nachträglich unveränderbar sind.

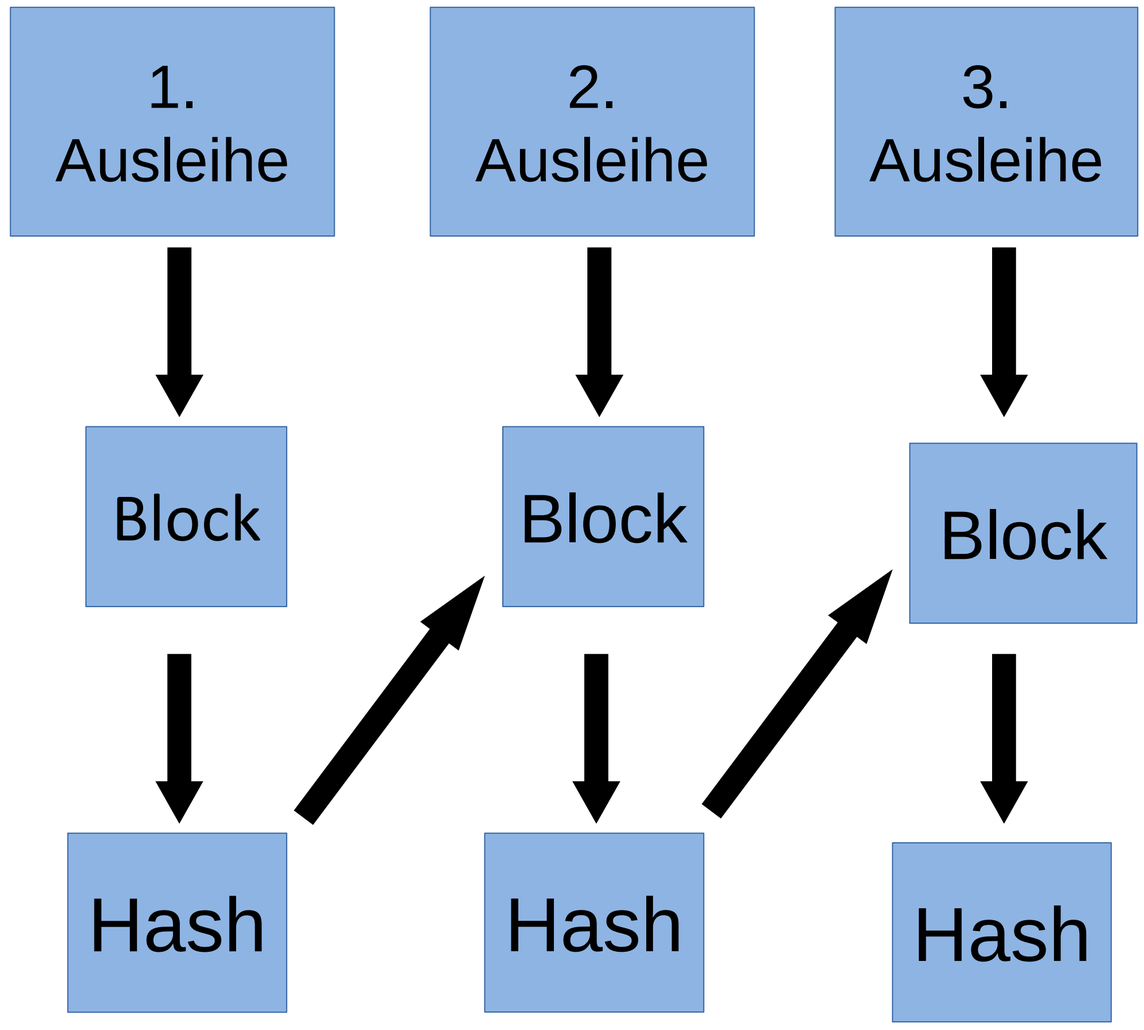
Erklärung des Peer-to-Peer-Systems

Die Sicherheit wird dadurch gewährleistet, dass das Netzwerk der Computer dezentral ist, es also keinen kontrollierenden Zentralcomputer gibt. Dadurch entscheidet immer die Mehrheit über die Richtigkeit der Daten. Dann kann ein Hacker nämlich nicht einfach über den zentralen Server alle Dateien verändern, sondern muss die Mehrheit der Computer hacken.



Erklärung der Blockchain

Eine Blockchain ist eine Kette aus Blöcken (Daten). Mit einem Hash-Algorithmus kann man den Hash für jeden Block berechnen. Jedoch wird für jeden Hash zur Berechnung auch der vorherige Hash verwendet. Dadurch werden alle Blöcke bis zum letzten verkettet – und deshalb muss man nur den letzten Hash beim Peer-to-Peer-System vergleichen.



Ergebnisse

Um die oben genannten Ziele zu erfüllen, haben wir ein Programm geschrieben, welches eine vorgegebene Datenliste (Blockchain) in einen finalen Hash umwandelt und ihn dann über ein Peer-to-Peer-Netzwerk mit allen anderen Computern vergleicht. Für den Austausch von Daten nutzen wir im momentanen Programm OneDrive, einen Cloud-Dienst von Microsoft.

Durch eine Blockchain und ein Peer-to-Peer-System gesicherte Online-Ausleihe

Wir haben versucht, eine Blockchain und ein Peer-to-Peer-Netzwerk zu programmieren, um damit eine sichere Online-Ausleihe für unsere Schulbibliothek zu erschaffen. Das sind neuartige Techniken, mit denen man Daten sicher zwischen verschiedenen Computern synchronisieren kann. Mit sicher ist gemeint, dass diese Daten „unhackbar“, also nachträglich unveränderbar sind. Das wird dadurch gewährleistet, dass das Netzwerk der Computer dezentral ist, es also keinen kontrollierenden Zentralcomputer gibt. Dadurch entscheidet immer die Mehrheit über die Richtigkeit der Daten. Dann kann ein Hacker nämlich nicht einfach über den zentralen Server alle Dateien verändern, sondern muss jeden einzelnen, oder zumindest die Mehrheit der Computer hacken. Ab einer gewissen Menge an teilnehmenden Computern wird dadurch ein extrem sicheres System erschaffen. Dieses wird Peer-to-Peer-Netzwerk genannt.

Um nicht alle Daten synchronisieren zu müssen, wird eine Prüfsumme aus den Daten berechnet, auch *Hash* genannt. Dieser Hash wird jedoch nicht einfach aus einem Datenblock berechnet, sondern aus allen. Dabei werden diese aufeinander aufbauend berechnet. D.h., für jeden Hash eines Datenblockes wird der Hash des vorherigen Datenblocks zur Berechnung genutzt. Wenn sich nun ein Eintrag im irgendeinem Datenblock ändert, ändert sich dadurch auch der dazugehörige Hash und durch die Abhängigkeit der Hashes voneinander auch alle darauffolgenden Hashes bis zum letzten. Deshalb muss man nur den letzten Hash mit den anderen Computern im Peer-to-Peer-System teilen - er ist der einzige nötige Wert zur Überprüfung der Korrektheit aller Datenblöcke.

Um die oben genannten Ziele zu erfüllen, haben wir ein Programm geschrieben, welches eine vorgegebene Datenliste (Blockchain) in einen finalen Hash umwandelt und ihn dann über ein Peer-to-Peer-Netzwerk mit allen anderen Computern vergleicht. Für den Austausch von Daten nutzen wir im momentanen Programm OneDrive, einen Cloud-Dienst von Microsoft. Dadurch ist das Programm nur so sicher wie OneDrive und der genutzte Microsoft Account. Um dies zu verbessern, wollen wir in Zukunft das schulinterne IServ-Netzwerk benutzen, dass alle Schulrechner miteinander verbindet. Für das Projekt haben wir einen Computer und einige Programmiersprachen (zuletzt „Julia“) benutzt.

Leider haben wir es bis zum Abgabezeitpunkt nicht geschafft, eine Funktion zum Hinzufügen neuer Blöcke fertig zu stellen. Hierbei liegt die Schwierigkeit darin, den neuen Block immer bei allen Computern gleichzeitig einzufügen, da sich durch einen neuen Block der Hash verändert. Bei einer Verzögerung haben dann nicht alle Computer den gleichen Hash und so werden einige fälschlicherweise gebannt. Allerdings haben wir uns bereits überlegt, wie wir dieses Problem lösen können über eine zusätzliche Datei, in der Benutzer Vorschläge für neue Blöcke eintragen können.