Project Proposal

Securing NoSQL databases using blockchains

MongoDB + ARK Blockchain

Jerpet, PB&J, FidgetAlferoSpinner

February 23, 2018

CS4411

Securing NoSQL databases using blockchains

Introduction (1-2 paragraphs)

The contents of a modern production database should be secure and interactions should be easily verifiable. We would like to propose a solution that strengthens the security of modern NoSQL databases using bleeding edge blockchain solutions.

In current database management software, the only way to verify the interactions that occur within a database is checking the log created by the system. The problem is that logging can easily be disabled or reduced to a level where malicious actions could occur without leaving a trace.

The solution we are proposing is to use hashing to capture the active state of a Mongo database and use an Ark blockchain to verify it's integrity. We would like to be able to specify a query resulting in a set of documents, the result will then be hashed and stored on the blockchain to a corresponding address. This functionality could then be automated to check the database's secured queries periodically to effectively secure it.

Motivation (1-3 paragraphs)

`What is the history of the problem?`

`Why is this problem interesting?`

`When and why does the problem occur?`

`Is the problem already solved? What is done now?`

`Are there any similar systems or solutions to the one you propose? If so, reference and very b`

`Are there are possible improvements to current solutions?`

`Where / how your solution will be used?`

Project Details (2-3 paragraphs design, 3-5 paragraphs deliverables)

We would like to design this project as a command line utility which allows the user to easily hash and validate the contents of a Mongo database running on the local machine. The project will consist of a Mongo database and a local deployment of an Ark blockchain. Unless complications come up with the platform, we are planning on using the Ruby language to develop the command line utility. Since we are using Ruby, we will be using the open source ark-ruby library to interact with the Ark blockchain and the official ruby mongodb driver to interact with the database.

The system will work by first prompting the user for a query to obtain a set of documents. The set of documents will then be hashed and a corresponding address on the Ark blockchain will be allocated for this query. The hash will of the set will be pushed in the form of a transaction to that specific address from a central blockchain management address (BCMA) which will act as one of the forging delegates and therefore always have funds to process the transaction. The transaction ID returned from the blockchain interaction will be stored back into the database in a control document storing the state of each secured set of documents so later on we can validate against this.

In this project we would like to deliver the functionality to manually make manually validate queries and detect tampering. The final deliverable will be the command line utility allowing the user to seamlessly interact with the entire system. If time permits, as an optional component, we would continue to integrate the validation system into the database and allow for automatic tamper detection. Another optional task which could be implemented in the future is the ability to secure all documents, not just the queries specified by the user. This project will contribute a system that is capable of utilizing the cryptographic power of blockchains to secure a locally stored traditional database. It provides a unique approach in between a fully blockchain based data storage and a NoSQL database.

Conclusion (1 paragraph)

Summarize the project including the problem, motivation, and proposed solution, and re-state important (planned) contributions.

References

List references used to compile proposal and references that will be used for project (if already known).