Project Proposal

Securing NoSQL databases using blockchains

MongoDB + ARK Blockchain

Gurpreet, Paul and Fernando

February 24, 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)

The motivation for combining these two technologies into a security solution comes from pre-existing database logging functionality. The common way to trace back the interactions that occur within a database management system is to look at its logs. Databases systems have the ability to disable logs or only record logs up to a specific error level. When relying on these logs to make sure no malicious activity has taken place, the investigator has to assume and hope that the logging is setup correctly and no one disabled it or tampered with it during the attack. This is where the problem arises; there is no completely secure way to verify the state of a database and ensure no one has tampered with the documents.

Databases provide fast and reliable data storage systems, and blockchain provide secure, distributed and immutable data storage. The problem is interesting because it has to potential to leverage the best qualities of both these technologies and improve data storage all together.

The idea of using a blockchain to secure the data stored in other types of storage systems has been attempted by a web service called Tierion using their token TNT. This system provides end points where you can send a hash and it sends back a timestamp ensuring it has been pushed to a

blockchain. A downside to this system is that it is a Software as a Service (SAAS) and therefore costs money to use on a large scale deployment. Private companies may not wish to have to send hashes of their data to an external service and Tierion does not allow for you to deploy your own instance of their system. The system itself is not very impressive either as it just provides a simple interface to the Bitcoin and Ethereum blockchains, neither of which are setup to handle such a task. Both BTC and ETH have long transaction times and were developed with a different purpose in mind. Tierion can take more than 10 minutes to push a hash to one of the chains which is too slow for an automated system. The idea is much better suited to data oriented blockchains such as the own we will be using called Ark. Ark has 7 second block times which will allow fast automated processing.

Ark is a Delegated Proof of Stake (DPOS) blockchain which allows transactions to carry a small amount of data. Each transaction has more than enough space to carry the hash of a single document. Using this free and open source blockchain also allows us to run a local instance which means we don't have to pay to secure our data and no information is exposed to the outside world.

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).