**Blockchain for Power Grids**

|  |  |  |  |
| --- | --- | --- | --- |
| Cadet Velez USMA,  Class of 2019  Company D4  Nicholas.Velez@westpoint.edu | Christian Banks  USMA, Class of 2019  Company B3  Christian.banks@westpoint.edu | Sam Kim  USMA, Class of 2019  Company F1  Sam.kim@westpoint.edu | Michael Neposchlan  USMA, Class of 2019  Company C1  [Michael.neposchlan@westpoint.edu](mailto:Michael.neposchlan@westpoint.edu) |

Abstract ---- **Sharing information is an important part of regulating and maintaining efficient and safe power grids. This project’s goal is to develop a way of using blockchain technology to share transaction information among different power grids in a secure, controlled, monitored, and efficient manner. The biggest concern regarding the data is non-repudiation. With the implementation of proper non-repudiation measures, the data will be reliable and attacks, such as man-in-the-middle attacks can be prevented. Proper accountability of the data is paramount. Hyperledger Fabric will be implemented in order to create a permissioned network in which power grids will act as nodes that maintain ledger information. By using a distributed ledger as a way to validate transactions though the process of consensus, the system will be able to share information in a manner that is more secure and transparent than traditional information sharing systems. The larger impact of such a system would allow for the further development of automated smart grids that can quickly and securely share information. These additional layers of security and speed would help to prevent issues, such as power grid failures, that could stem from the latency involved with traditional methods of validating, processing, and reacting to shared data.**

**Index Terms - Blockchain, hyperledger fabric.**

1. INTRODUCTION

Sharing information across large networks poses many issues and potential risks involving security and usability. For the purpose of sharing information between microgrids, security and confidentiality are paramount. Blockchain architectures provide a variety of solutions to sharing information. Of the many different blockchain solutions currently available, Hyperledger provides superior scalability, modularity, and security [1]. For this project, in order to facilitate the sharing of information among various microgrids in a way that will potentially be scalable to much larger industry grids, a hyperledger fabric network will be implemented.

1. BACKGROUND

a. *Blockchain.* Blockchain is a group of transactions that are linked to their previous modification on a specific channel.[2] The chain is a log that contains the transactions of all previous ‘blocks’ for that particular chain. When a new block is appended unto the chain, the transaction from that previous block is also appended.[2]

A large feature of a blockchain is the use of a distributed ledger.[2] Blockchain ledgers are often decentralized because each person on the network is working with their own replication of the block.[2] Utilizing a decentralized ledger helps add security to the network as all information is not funneling into one node. The ledger contains two characteristics, a world state and a blockchain.[3] A blockchain was explained in the previous paragraph, however, a world state is a database that keeps the values of a ledger state at the current time.[3] This allows anyone or program to pull the current state at any amount of time without having to sweep a log.[3]

b. *Hyperledger fabric.* Hyperledger fabric is a type of blockchain that involves a collaborative approach to community sharing, property rights and the development of standards over time.[5] Just like other blockchain projects, Hyperledger has a ledger and utilizes smart contracts.[5]

1. DESIGN

a. *Powergrid Network.* The endstate for this project is a network with connectivity across six peer nodes including: the United States Military Academy (USMA), the United States Naval Academy (USNA), the United States Air Force Academy (USAFA), Army Research Labs (ARL), Iowa State University (ISU), and Idaho national labs (INL). In essence, all of these nodes will represent microgrids that can communicate with each other and share data in a secure way on the network using Hyperledger. The validating entities inside of the network are USMA, USAFA and ISU. The validating entities will validate transactions, maintain the ledger while ensuring consensus has been achieved on all transactions. Appendix A, Figure 1 is an example of the final network design in which all six peer nodes will reside.

b*. Internal Network.* The process of data retrieval from the node network is explained in our internal network diagram found on Appendix A, Figure 2.

MySQL data from the USMA MicroGrid will be read by one of the hosts located at USMA. The data from that PC will be read by the database OPVN client (Node). Data is obtained by running a database query algorithm. This algorithm which is explained in the next paragraph. Then from the OPVN client, data will be sent out the VPN concentrator with a update request to the rest of the nodes on the network, and the nodes will be able to confirm the request to update the ledger. The other nodes will be able to connect to the VPN concentrator in order to send confirmation requests back to our client. Once the request is sent back to our client node validating the update, the ledger state is updated with the appropriate information. Each node on the network will be running a similar instance of OPVN client node as seen on Appendix A, Figure 2.

c. *Database Query Algorithm.*The python script first establishes a connection to the MicroGrid SQL database using proper authentication. After successfully connecting, the script is read from a constraints file provided to by the Real-Time Smart Grid Capstone team. This file contains the IDs of all the PMUs and applicable variables necessary for queries. The program then loops through each PMU and queries it for the latest readings that fall within the established minimums and maximums. The queries are written to a file in readable format. Additionally, each query is printed to the console (for verification the script is working) and finally closes all files and connections. In order to more easily create historic data for our network, the script will be updated to accommodate JSON formatting.

d. *Scalability*. Once a successful proof of concept can be run, our framework can be adapted on a larger scale to allow peer nodes to communicate over large distances. Latency would be affected by the changes in the environment. However, as the algorithm improves and is optimized, these effects can be mitigated. Appendix A, Figure 5 shows the 6 microgrids that will be part of the blockchain. As observed on the figure, latency will be an issue as there is a large distance between the microgrids in Idaho, Colorado, and Iowa and the microgrids on the east coast. With larger amounts of latency, the potential for the control system to become unstable rises. This is something that will be accounted for when planning for expanded scalability.

1. IMPLEMENTATION

*a. Database interface*. The data that will be shared in this project is collected from Phasor Measurement Units (PMUs) connected to the local power grid. These units collect voltage and current measurements from different parts of the microgrid. Out algorithm queries the database and returns the relevant data in a timestamped and shareable format.

*b. Peer Node*. For the peer node network, we are currently implementing Hyperledger architecture in a virtual environment. Two peer nodes are established and can connect to one another. On the client node, there is an orderer terminal, a command line interface, and a log file being updated in the background. On the manager node, there is a certificate authority server, an orderer, and a Couchdb server. The Couchdb server allow for the database to be viewed through a web interface.

1. RESULTS

a. *Building two nodes.* Our team adapted a tutorial in order to create a two node hyperledger architecture utilizing virtual machines to successful conduct a hello world test. In order to verify the communication amongst the nodes, a CouchDB server was utilized to record all relevant information - which was encrypted through hashing.

*b. Data Retrieval*. Through python coding, we were able to successfully pull data from the microgrid at West Point. The data that we are working with is: wattages, currents, voltages, and time stamps exiting the microgrid. Figure 6 is an example of a successful pull from the West Point microgrid. Our code is modified to display what PMU we are accessing, the value of the voltage and the time stamp ID (seen as tsmID in Fig. 6). The voltage reading that displays none, means that the PMU has not been recording or has no data for that timestamp. Our next goal is to use javascript code to inject our script into the hyperledger network that we created.

1. FUTURE WORK

a. *End of the Project*. The end of the project is marked by the successful communication with other microgrids under a secure hyperledger network. Additionally, the latency will be optimized to ensure that data is being transferred and received at the fastest speed possible without bearing a large load on the system.

b. *Future Work*. The implications of this technology apply to a global level of communication amongst power grids that rely on one another to supplement power when needed. A hyperledger architecture supports a secure method of communication in an infinitely large network. This process expedites the sharing of information which results in the prevention of potential blackouts relevant to power grids.

1. Conclusion

a. In this paper, the basics of blockchain and hyperledger fabric are explained to help give understanding to the project: Blockchain for powergrid. First, the design of the powergrid network, and internal network, database query algorithm and scalability is discussed. The implementation of our system and the current results were also discussed. Lastly, recommendations for future work work were given, such as upscaling our project past communication between schools and movement to communication between larger scale power grids. By implementing our design we can create a system that allows microgrids to communicate relevant data across many peer nodes. This can help solve issues that stem from the current information sharing systems.

Appendix A

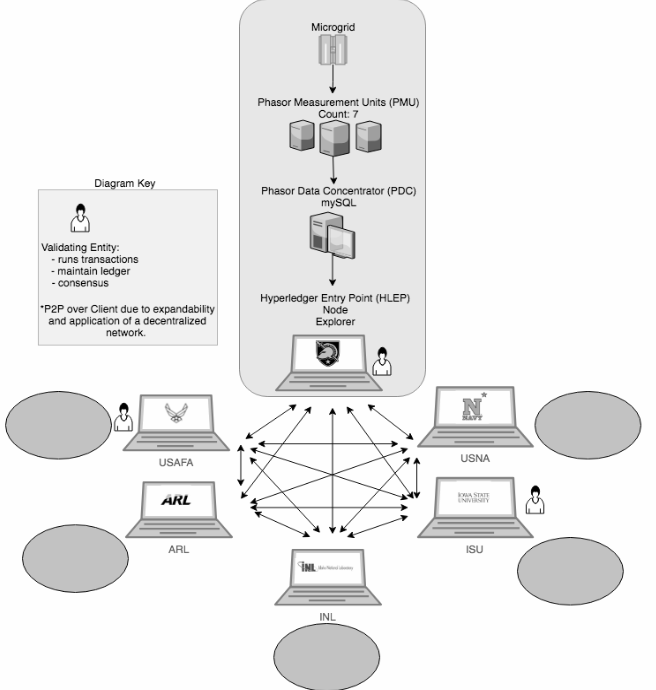


Fig. 1

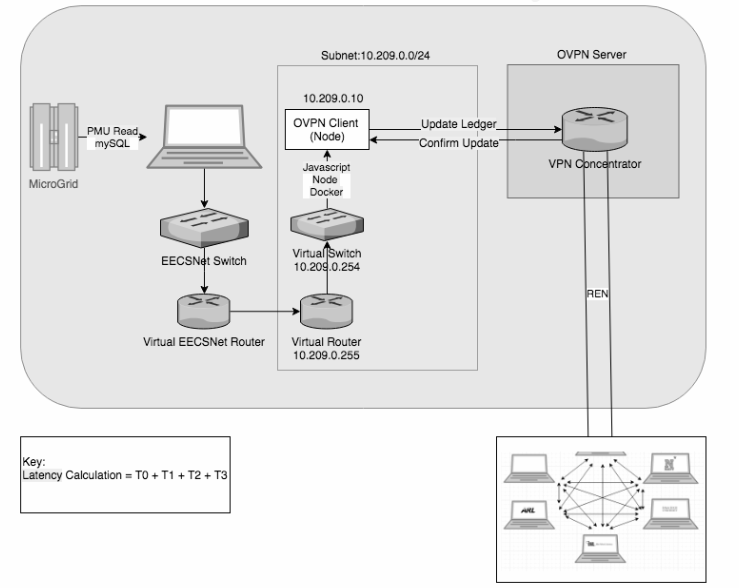


Fig. 2

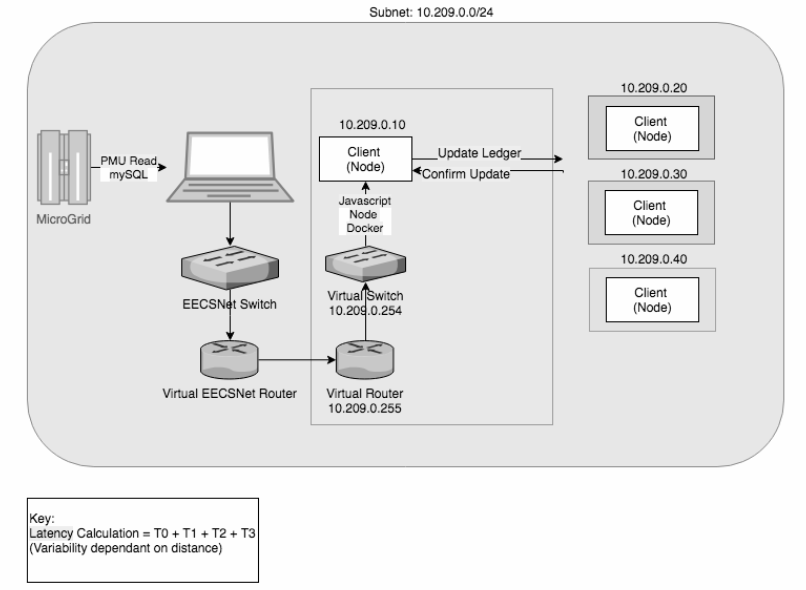


Fig. 3

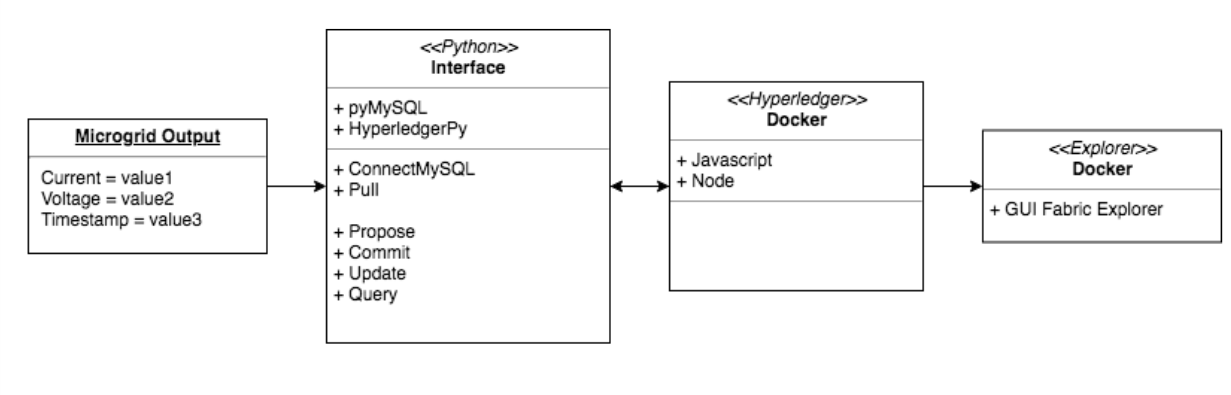


Fig. 4



Fig. 5

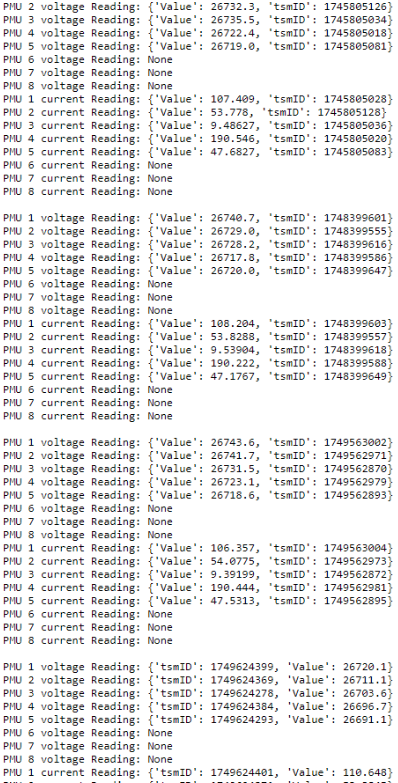


Fig. 6

References

1. <https://www.hyperledger.org/projects/fabric>
2. https://hyperledger-fabric.readthedocs.io/en/release-1.2/glossary.html
3. https://hyperledger-fabric.readthedocs.io/en/release-1.2/ledger/ledger.html
4. <https://trustindigitallife.eu/wp-content/uploads/2016/07/marko_vukolic.pdf>
5. <https://hyperledger-fabric.readthedocs.io/en/release-1.2/blockchain.html>