N232-094, Blockchain-based, Highly Secure, Decentralized, and Immutable (DSI) Network System Protocol for Multifunction Advanced Data Link (MADL)

Responses to questions posted to DSIP Topic Q&A

1. All limitations depend on the communication environment. Your Blockchain like solution should be an open-source software-based platform and independent of the hardware platform. Your Blockchain like solution should work on generic platforms, like X86 and ARM. In terms of running environment, it can work on native OS environments Linux, Windows, and Mac, or docker containers supported by the underlying OS. Blockchain-based communication adopted by Blockchain like solution should work at the application protocol level atop of TCP/IP (level 3). Thus, your Blockchain like solution solution should not be limited by level 2 (data link) and level 1 (physical) communication. Since MADL works at Ku-band fast switching narrow directional communications, your Blockchain like solution should work without major modification

Hardware or Software for your review and consideration:

- Size, Weight and Power and Cost (SWaP-C) – 1) Neuromorphic processing – Intel Loihli hardware, LAVA software; 2) Raspberry Pi 4 Model B boards as edge devices with ZPiE open-source library. ZPiE: Zero-knowledge Proofs in Embedded systems, GitHub, https://github.com/xevisalle/zpie; and 3) NVIDIA Jetson modules

- Interoperable with NSA-approved Commercial Solutions for Classified (CSfC) solutions that use two layers of commercial encryption.

- The Inter-Blockchain Communication Protocol (IBC) is a protocol to handle authentication and transport of data between two blockchains. IBC requires a minimal set of functions, specified in the Interchain Standards (ICS). https://github.com/cosmos/ibc

- Secure Spectrum Sensing based on Blockchain (SSSB) algorithm to evaluate the reliability of nodes in real-time and improve energy efficiency and sensing performance of cognitive wireless of MADL.

- Ad Hoc On-Demand Distance Vector (AODV) routing protocol

C. Ran, S. Yan, L. Huang, and L. Zhang, “An improved AODV routing security algorithm based on blockchain technology in ad hoc network,” EURASIP journal on wireless communications and networking, vol. 2021, no. 1, pp. 1–16, 2021.

2. Your Blockchain like solution should follows the principles of permissionless network management. Any regulatory and compliance rules by permissionless management should also be applied. Permissionless blockchain models are founded on the idea that the nodes in the network do not trust each other and so every node in the network maintains its own identical copy of the ledger and is responsible for checking the validity of transactions on its own.

For your consideration and possible use the following National Institute of Standards and Technology (NIST) information is provided in answering the question:

For general encryption, NIST has selected the CRYSTALS-Kyber algorithm. Among its advantages are comparatively small encryption keys that two parties can exchange easily, as well as its speed of operation.

For digital signatures, often used to verify identities during a digital transaction or to sign a document remotely, NIST has selected the three algorithms CRYSTALS-Dilithium, FALCON and SPHINCS+ (read as "Sphincs plus"). Reviewers noted the high efficiency of the first two, and NIST recommends CRYSTALS-Dilithium as the primary algorithm, with FALCON for applications that need smaller signatures than Dilithium can provide. The third, SPHINCS+, is somewhat larger and slower than the other two, but it is valuable as a backup for one chief reason: It is based on a different math approach than all three of NIST's other selections.

Three of the selected algorithms are based on a family of math problems called structured lattices, while SPHINCS+ uses hash functions. The additional four algorithms still under consideration are designed for general encryption and do not use structured lattices or hash functions in their approaches.

Dilithium is a component of the CRYSTALS (Cryptographic Suite for Algebraic Lattices) suite that was submitted to NIST's call for post-quantum cryptographic standards. It is a digital signature scheme that is strongly secure under chosen message attacks based on the hardness of lattice problems over module lattices. The security notion means that an adversary having access to a signing oracle cannot produce a signature of a message whose signature he hasn't yet seen, nor produce a different signature of a message that he already saw signed. For the same security levels, Dilithium has a public key that is 2.5X smaller than the previously most efficient lattice-based schemes, while having essentially the same signature size.

Reference:

1] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler and D. Stehlé, "CRYSTALS-Dilithium: A Lattice-Based Digital Signature Scheme," ACR Transactions on Cryptographic Hardware and Embedded Systems, vol. 2018, no. 1, p. 238–268, 2018.

2] G. Alagic, J. M. Alperin-Sheriff, D. C. Apon, D. A. Cooper, Q. H. Dang, C. A. Miller, D. Moody, R. C. Peralta, R. A. Perlner, A. Y. Robinson, D. C. Smith-Tone and Y.-K. Liu, "Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process," NIST, 2019.

3] Dilithium - CRYSTALS https://pq-crystals.org/dilithium/software.shtml

A possible authentication process could be performed by generating One-Time Pad (OTP) with a pseudo random function. The nodes could be registered in the Blockchain, and each node determines the nearest node that it is able to authenticate based on the relationship stored in the Blockchain nodes. The request for authentication is sent from a node to the related node which observes and checks in the Blockchain whether this node is related and would be able to authenticate it. This scheme is able to thwart the attack of external malicious nodes or third-party attacks, even if the adversary knows the first token.

3. If your Blockchain like solution requires a synchronous network environment then it is sensitive to network latency. Network delays and packet loss may influence performance. Recommend using fifty nanoseconds for a one data exchange from Node A to Node B.

Consider setting up a distributed, multi-simulation toolchain, for your Blockchain like solution analysis and anomaly detection using Ku-band signal data reporting which would enable:

• Blockchain like solution prototype design with edge-fog-cloud platforms;

• Seamless integration with security microservices;

• Enable data analyzing services towards feature extraction from airborne data (Ku-band);

• A multi-functional user-friendly front-end GUI;

• Message queuing service for acknowledgment between front-end GUI and back-end service

• Kafka streaming service for real-time avionics digital data synchronization

• Data points to be recorded are

- nMessage: Numer of exchanged messages to achieve consensus
- nodeDelay: Processing delay at node level required for achieving consensus.
- nodeConsen: Number of participating nodes required to achieve consensus
- consenDelay: Network delay for the consensus protocol to achieve consensus
- consenPauseState: Probability that the consensus state is paused for a special case
- pauseStateTime: Time that the consensus system remains in the paused state

4. All data (voice, text, imagery and video) used by your Blockchain like solution should be transmitted on a Ku-band as bytes or binary strings, no specific data types and formats are needed by communication protocols. Your Blockchain like solution should operate in a bandwidth that satisfies the common Ku-band communication standard (e.g., 548 Mbps upload and 1 Gbps download speeds).

For your consideration and possible use the following information is provided in answering the question:

Immutable datatypes are objects that cannot be modified or altered after they have been created (for example, by adding new elements, removing elements, or replacing elements). Python's immutable data types are:

• Int.

• Float.

• Tuple.

• Complex.

• String.

• Stringfrozen set [note: immutable version of the set]

• Bytes.

When you make changes to immutable objects, the memory where they were stored during initialization is updated.

Quality of Service

• Frequency of information updates: the rate at which updated values are sent or received.

• Priority of data delivery: the priority used by the underlying transport to deliver the data.

• Reliability of data delivery: whether missed deliveries will be retried.

• Parameters for filtering by data receivers: to determine which data values are accepted and which are rejected.

• Duration of data validity: the specification of an expiration time for data to avoid delivering "stale" data.

• Depth of the 'history' included in updates: how many prior updates will be available at any time, e.g., 'only the most recent update,' 'the last n updates,' or 'all prior updates'

Quality of Information (QoI)

QoI Metric	Trust In	Metric Input
Pedigree	Chain-of-custody	Parent document metadata
Provenance	Original sources	Reputation of authors
Reputation	Publisher	Reputation of publisher
Correctness	Accuracy of data	SOA service opinion
Truth	Validity of data	Human vetting
Timeliness	Freshness of data	Expiration date/time
Data Integrity	Data has not been modified	Digital signatures/certificates
System Integrity	Underlying system	Configuration file
Disclosure	Authorized dissemination	Reputation of source
Relevance	Receipt of necessary info	Subscription metadata
Corroboration	Supporting authors	Reputation of authors
Compliance	Structure of data	Schema

5. The Bank Secrecy Act (the "BSA") does NOT apply to this SBIR topic. The Financial Crimes Enforcement Network (FinCEN) is NOT part of this SBIR topic.

6. For your consideration and possible use the following is provided in answering the question:

• PostgreSQL-based baseline system which should include: Network-level security (e.g. unauthorized network connections); Transport-level security (e.g. Client certificate authentication); and Database-level security (e.g. authorization or access controls).

• Pumba tool to simulate bandwidth degradation and disconnection between different parties on the network. Alexi Ledenev, "Pumba: chaos testing tool for Docker (Github)," Pumba: chaos testing tool for Docker (Github). https://github.com/alexei-led/pumba (accessed Jan. 06, 2021).

• The eXtensible Access Control Markup Language (XACML) is a standard that defines a fine grained attribute-based access control policy language. There are open-source XACML evaluation engines written in popular programming languages that you can leverage as opposed to creating our own system.

• Automated Validation of Internet Security Protocols and Applications (AVISPA) tool. It is a security protocol verification and analysis tool in which new security protocol has to be written in High Level Protocol Specification Language (HLPSL) and then fed as input to the tool. AVISPA consists of four different back-end compilers: On-the-fly Model-Checker (OFMC), Constraint-Logic-based Attack Searcher (CL-AtSe) , SAT-based Model-Checker(SATMC) and Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP).

• Java Native Interface (JNI) native language allowing interoperability between Java and C

• Python implementation framework - Python Multiparty Computation Implementation

• Multiparty Computation (MPC) protocol by Smart, Pastro, Damgård and Zakarias

• IPv6 for mobile devices such as drones because there is no need to go through a network address table (NAT) and thus is low latency. IPv6 will be combined using blockchain with Proof of Stake consensus

• Programming language: Python, JavaScript, and bash scripting.

• Development: OS (Ubuntu 20.04), Hardware (Desktop and Raspberry Pi).

• Tools: IDE (VS code), Container dev (Docker), Putty or other SSH tools.

• Bitmessage Plus - J. Warren. (2012). Bitmessage: A Peer-to-Peer Message Authentication and Delivery System. [Online]. Available: https://bitmessage.org/bitmessage.pdf

• Do not use any blockchain that supports Turing-complete on-chain execution (e.g., Ethereum, Hyperledger, and Tezos) because those blockchains cannot enforce semantic immutability.