IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 22 March 2024, accepted 17 May 2024, date of publication 20 May 2024, date of current version 28 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3403483

RESEARCH ARTICLE

A Zero-Trust Satellite Services Marketplace Enabling Space Infrastructure as a Service

GREGORY FALCO, (Member, IEEE), AND NATHANIEL G. GORDON
Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14850, USA

Corresponding author: Gregory Falco (gfalco@cornell.edu)

This work was supported by the Defense Advanced Research Projects Agency.

ABSTRACT Exponential growth of the space industry avails unprecedented opportunities to establish a marketplace of satellite infrastructure services. However, security and resource constraints pose critical challenges to implementing the exchange of services such as storage, compute or even arm-based manipulation. We propose a fully distributed architecture that will facilitate resilient, trustless interactions to enable space infrastructure as a service and applications such as in-space servicing. The distributed architecture engages Distributed Ledger Technology (DLT) such as blockchains and directed acyclic graphs to designate and enforce security policy via smart contracts between multiple parties across payloads owned or operated by different service providers on the same satellite bus or across a constellation. This work presents a zero-trust space infrastructure as a service architecture and examines how the architecture addresses critical challenges such as consensus and cyber resilience to facilitate a space services marketplace.

INDEX TERMS Space cybersecurity, zero-trust, space architecture, space technology, space cloud, distributed ledger technology, directed acyclic graphs, blockchain, smart contracts, resilient space systems, infrastructure as a service, in-space servicing assembly and manufacturing.

I. INTRODUCTION

To truly scale operating capacity, the space sector must evolve from the traditional, vertically integrated space vehicle development and operations model to facilitate the exchange of services across multiple payloads owned and operated by different parties. While there is commercial and government interest in enabling infrastructure as a service capabilities, there are functional challenges - the principal being a matter of security. To provide services between two payloads (either on the same bus or across vehicles) that do not inherently trust each other, a moderator is required to enable a fair interaction. However, a centralized moderation capability would present a ripe target for an attacker, eliminating any semblance of a zero-trust exchange.

We propose a fully distributed architecture that will facilitate resilient, zero-trust interactions to enable space infrastructure as a service (SIAAS) called the Orbital Resilient Blockchain Interagent Transaction Service (ORBITS) architecture. One particular area of interest for this architecture is for In-Space Servicing, Assembly, and Manufacturing (ISAM) missions, where two or more space assets must physically interact to exchange a service. The distributed architecture engages a blockchain to designate and enforce security policy between multiple parties across payloads owned or operated by different service providers on the same bus or across a constellation. The blockchain-enabled architecture combined with a physics-based validation provides a consensus mechanism to identify bad actors across the architecture and is resilient to node failures or attacks. Policy-defining interactions that are delivered over the blockchain provide the benefits of a software-defined network, with the resilience afforded by distributed ledger technology (DLT).

The ORBITS architecture is distinct from current approaches to enable shared services. Today, should two payloads wish to share services such as graphics processing units (GPUs), there are no means for mediated interaction. There is either open communication over the bus or between two space vehicles or no engagement at all. Virtual machines (VMs)

The associate editor coordinating the review of this manuscript and approving it for publication was Juan Liu.

are used to provide isolation between payloads, enabling some security and enhancing resident software isolation from other environments. While hypervisors can help to manage VMs, they do not facilitate a trustless exchange of information between them, nor can they enforce policies governing integral interactions across the VMs. There is interest in engaging across payloads using homomorphic encryption; however, this is a slow and compute-intensive process that is not viable on a space vehicle today.

Establishing a trustless and resilient mechanism for payloads owned by different organizations to engage with each other to perform in-space servicing could be transformative for the sector. Today, ISAM missions such as NASA's OSAM-1 or DARPA's RSGS are arranged years in advance, and are performed between trusted parties. A proliferated network of service-capable spacecraft linked by secure infrastructure could enable *ad hoc* servicing between unaffiliated parties. Moreover, rather than building out vertically integrated space assets, startups can focus on niche capabilities while relying on payloads, developed by other organizations, for supporting services, thereby reducing the barrier to entry to the space sector. Commercial organizations can offer revenue-generating services to others requiring infrastructure such as compute, storage, and sensors, without the concern of being harmed in the process. Drastic cost reductions are possible for developers and space operators if they can securely rely on others for service. Similarly, defense and intelligence agencies can reduce their operating costs and increase their launch and operating capacity by relying on the private sector to provide support services. Ultimately, similar to how cloud services reduced the barrier to entry for organizations wishing to engage with resource-intensive procedures, trustless, resilient, facilitated mechanisms to enable SIAAS will similarly benefit the space industry.

This paper outlines the existing landscape of distributed space infrastructure and associated services, describes the current state of space asset service exchange technology, defines design parameters for zero-trust engagement, and establishes an architecture to facilitate a zero-trust engagement across assets. The paper concludes with potential limitations of the approach and current development efforts. The principal technical contribution is the introduction of a resilient architecture and supporting abstract protocol for enabling DLT-backed, integral service sharing between critical space assets.

II. PRIOR ART

Here we describe inter-satellite communication technologies, the distributed space system technology landscape and distributed ledger technologies. Further, we describe a variety of DLT projects relating to the space sector and the challenges of engaging DLT for space vehicles.

A. SPACE LINKS

Before space systems are able to exchange services, reliable and secure links between assets will be required. While space links have existed for decades, researchers, government organizations and industry are still developing and refining media for space-related communications. Today, communications from space vehicles to other assets - be they ground stations or other space vehicles, are designed to be received by trusted agents, usually over an encrypted link. However, the assumption that data sharing is only necessary among trusted parties is flawed. Future space vehicles will be highly interdependent and require a means for transacting data and services with others to enable seamless operations. Figure 1 illustrates existing Global Navigation Satellite Services (GNSS) and Communication satellite constellations whose owners claim cross-link capability within their own constellation.

While radio frequency links can be used between space vehicles, optical signal offers an alternative means of communication. Optical communications offers a number of benefits compared with RF including high data rates, precision pointing, license-free spectrum and large bandwidth; however, atmospheric affects including absorption and scattering degrades the integrity of these links [9]. Such interference results in degraded trust and transparency of operations across space systems. Optical links are further challenged by the need for optical receivers to be appropriately configured to integrate with a variety of signal.

Integrating the two communication modalities across a multi-agent space network could help to leverage the strengths of each, while mitigating their challenges. Space-Based Adaptive Communications Node (Space-BACN) is a DARPA program, envisioned by program manager Greg Kuperman, that aims to establish a space internet across distributed networking assets and constituent participating nodes. Space-BACN can transform the sector by enabling a communication ecosystem with heterogeneous assets; however facilitating zero-trust exchange of services will persist as a gap that could inhibit the desire for space systems to interact.

B. DISTRIBUTED SPACE SYSTEMS

Distributed function across space vehicles is not a new concept. As early as 2006, Brown and Eremenko described technical uncertainties and reliability concerns relating to monolithic spacecraft and the value in distributing functionality across independent modules that interact wirelessly [10]. They called this approach to spacecraft development ''Fractioned Space Architectures'' which was explored as part of DARPA's System F6 program [11]. A series of studies transpired as a result of this program about the value proposition of a digitally enabled and fractioned system [12], [13]. System F6 program yielded a F6 Development Kit which included an open source software supporting the physical wireless link layer for communications across modules [14]; however, no further reports of development were released thereafter.

Since then, fractioned architectures have been further explored including concepts such as the ''Federated Satellite

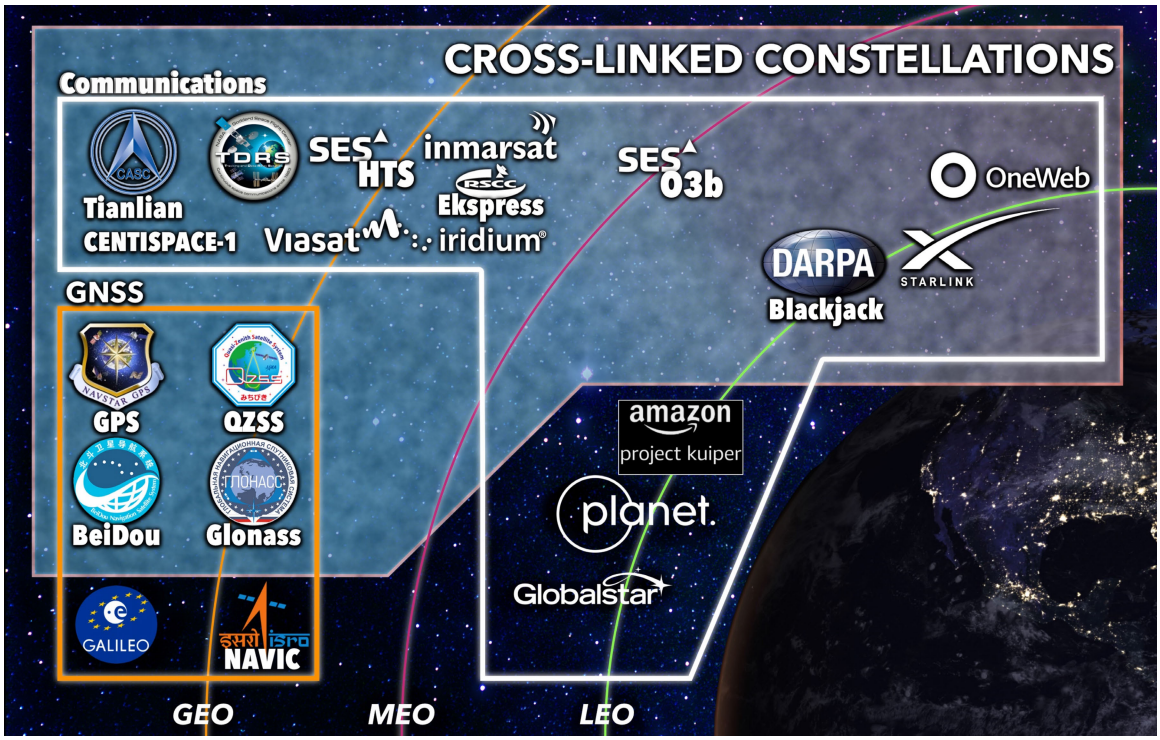

FIGURE 1. Constellations with crosslink capabilities [1], [2], [3], [4], [5], [6], [7], [8].

System" [15]. Financial value propositions for constituent architectures that described the cost efficiencies of engaging with IoT-like components across distributed ecosystems of spacecraft were further described through business cases, similar to studies for System F6 [16]. Subsequently, a research agenda for distributed satellite systems was published describing the persistent limitations to implementing fractioned space architectures and notably calls for the application of blockchain across systems to facilitate agent interactions [17].

C. DISTRIBUTED LEDGER TECHNOLOGY COMMUNICATIONS

Distributed ledger technology (DLT) could help exchange information and services in a resilient way between space vehicles and their interdependent assets.

DLT consists of a distributed, shared and synchronized database that exists across geographies. There are both permissioned and permissionless ledgers. Permissioned ledgers are distributed and synchronized, but access to the system is controlled by a single administrator. This generally mans that permissioned DLTs are smaller and inherently more private given there are access controls. Permissionless DLTs are open to the public where anyone with a server and the appropriate software can participate [18].

1) BLOCKCHAIN

The most popular DLT is a blockchain, which was introduced in 2008 by Satoshi Nakamoto in a white paper detailing the operations of Bitcoin, a peer-to-peer currency exchange system [19]. A blockchain captures incremental data transactions in "blocks" that are confirmed by distributed nodes called miners. Consensus across the miners is required before data can be hashed on to the blocks which are connected in a chronological chain after being identified by a cryptographic hash.

Blockchains are considered to be resilient to attack and of high integrity as the compromise of any single node or collection of nodes will not be sufficient to change the history recorded by all the distributed nodes across the chain. 51% of the processing power commanding the miners of a blockchain would need to be compromised to achieve what is called a "double-spend" attack, where recently hashed blocks could theoretically be re-allocated. The size and hash rate of large blockchains such as Bitcoin makes the 51% attack extremely unlikely to occur given the massive amount of computing power that would be required to take over 51% of the mining power of the chain. This is less true of smaller blockchains, often called altcoins, that have previously experienced such attacks [20].

Bitcoin is a permissionless blockchain, as is Ethereum (first described in 2013) - a blockchain whose purpose is to enable contracting between parties via "smart contracts" [21]. There are several examples of permissioned blockchains in production such as Hyperledger which was started in 2015 [22]. Permissioned blockchains are engaged when there is an agreed-upon central authority, whereas permissionless chains are best for transactions that seek no authoritative governor

of who can partake in the transactions and visibility of the ledger.

Blockchains such as Bitcoin and Ethereum require a considerable amount of memory for the storage of blocks as well as processing power in order to mine each transaction and hash it onto the chain. This has presented challenges for engaging blockchain on constrained IIoT devices that could otherwise benefit from the integrity features previously described. Several modifications of both the Bitcoin and Ethereum blockchain exist which aim to preserve the benefits of these large chains, while enabling operations on constrained devices [23], [24].

2) DIRECTED ACYCLIC GRAPHS

Another type of DLT is a Directed Acyclic Graph (DAG). They are far less popular than blockchains, but also offers benefits associated with distributed ledgers. Instead of forming blocks where consensus is required across the majority of nodes, transactions are written on the ledger and then must validate two other unvalidated transactions before being posted. The transaction then must be validated by newer succeeding transactions. The benefit of a DAG is that it does not require the extensive processing required to hash transactions to the chain or the memory to replicate the chain on every distributed ledger [25].

D. DLT OPPORTUNITIES FOR SPACE VEHICLES

Given the communication integrity challenges described for RF and optical communications, a resilient and immutable distributed ledger of communications to be shared across distributed space vehicles could be highly desirable. Various use cases and simulated experiments have been conducted to demonstrate the applicability of DLT to space systems.

While there are not many publicly discussed DAG projects relating to space vehicles, a DARPA program called System F6 engaged a series of small satellites to collaborate within a constellation using a DAG [26]. The DAG enabled real-time workload allocation and collaboration among the trusted nodes. The project was used as a model for other distributed real-time applications [27].

Blockchain space use cases and experiments are more common. A series of experiments were developed for space systems using Hyperledger and Ethereum by Mital et al. The team's experiments included logging and tracking command and control events, command provenance and record keeping, encrypted command transmission, command transmission to multiple, specific entities, and acknowledgement transmittal [28].

Such experiments were followed by Xu et al. describing how a modified Ethereum blockchain could be engaged for space situational awareness. They developed an access control scheme that allows a variety of actors to achieve secure identity authentication for certain information within ''virtual trust zones'' [29].

Feng and Xu subsequently developed an blockchain-embedded satellite communications delay-tolerant network. The network was designed to both help detect satellite communication cyber attacks and enable resilient communication in light of an attack [30].

Despite these use cases and experiments, there are few planned blockchain projects for space systems. One project includes Xage Security's contract with the United States Space Force announced in September 2020 that plans to use blockchain for verified access control of space communications [31]. Another project called SpaceChain was funded in 2019 by the European Space Agency to develop an open-source satellite network where each satellite will be a node contributing to the distributed ledger [32]. SpiderOak, an authentication services provider that engages private blockchains, won a contract from the Defense Innovation Unit (DIU) to demonstrate its capabilities in orbit [33].

E. DLT CHALLENGES FOR SPACE VEHICLES

Despite the interesting and ostensibly valuable use cases and associated experiments described, each present implementation challenges enumerated by the research teams. There are technical, economic and social barriers to adoption. The challenges likely contribute to the poor translation of evolving space DLT concepts from research to practice.

A technical challenge with DAGs is that because transactions are validated by others wishing to make transactions, rather than miners who are paid to validate transactions, there is no incentive structure for a transaction to select any particular other transaction for validation. Some DAGs use a centralized coordinator that directs new transactions towards others that need to be validated. This could cause a centralized security risk for DAGs [23].

Two technical challenges persist with blockchain. The latency inherent in transaction validation is a primary concern [28], [29]. For certain scenarios, especially those that are safety-critical, near real-time processing and broadcasting is required. In addition to latency issues, the memory-constrained nature of space vehicles present challenges over time given that as transactions increase in number, the ledger will grow. Given all transactions are permanently stored on all nodes of a blockchain, the memory constraints of space vehicles will present an issue in maintaining the complete blockchain nodes [28], [29].

While not explicitly noted as a challenge in the literature, recent costs of cryptocurrencies such as Ethereum make blockchain technologies less attractive from an economic utility standpoint. This cost is attributed to gas (also called wei) that is paid to the miners and based on the current price of the cryptocurrency. Cryptocurrency prices are highly volatile resulting in potentially high transaction costs. While Ethereum is not the only blockchain available to use, each have a cost. Because cryptocurrency assets are so volatile, a risk is that services provided by a major blockchain could become prohibitively expensive at scale.

Finally, there are social barriers to engage with blockchain. Jones describes DLT as a ''foundational technology'' which may find adoption sooner in some industries than others. The space sector may be less conducive to embracing DLT given the sector's unease with a void of a central authority. Stakeholder involvement, coordination, legal and regulatory acceptance and standards - along with a series of isolated proof of concepts, will be needed to drive widespread social adoption of DLT in the space sector [34].

Given the myriad challenges bringing DLT for space systems from the lab to space vehicles, a distinct approach is required that builds on the technical capabilities of blockchain that enables trust and coordination, but is customized for the nuances of space vehicles and the safety critical systems they support.

F. APPLICABILITY OF DLT TO ON-ORBIT MISSIONS

In-space Servicing, Assembly, and Manufacturing (ISAM) refers refers to a group on in-space operations where one spacecraft repairs or creates another. These mission types bring the possibility of greatly extended mission lifetimes and lower operational costs, but require a significant investment in on-orbit infrastructure to realize [35] . However, as a critical mass of space assets with servicing capabilities becomes prevalent, a unifying platform for requesting and exchanging services is necessary [36]. The integrity of service exchange for this application is of particular significance due to the potential for a servicing operation to inflict harm, either through software faults or intentional sabotage, during proximity operations. This concern is especially heightened if servicing is being performed between two unrelated stakeholders.

III. CURRENT STATE OF SPACE SERVICE EXCHANGE

To provide shared services across a single bus, payloads must communicate with each other, but there are no means to achieve this without integrity concerns today unless interacting payloads are all developed or operated by the same party. When available, security for interagent interactions is centralized where there is a single application that has supervisory control over the exchange. Centralized security is imperfect because a malicious actor could corrupt the central supervisor, rendering the exchange compromised. Other security approaches within the same bus include hosting payloads in different virtual machines (VMs), thereby isolating each payload. Hypervisors may afford the central management of multiple VMs, but again, this could become compromised which limits the extent of security provided by the VMs. Commanding multiple payloads in a given hypervisor across vendors would require prior planning and trust vetting by the payload developers and cannot be configured on-the-fly when service across payloads is needed.

Today, there are no secure means for satellites across constellations to share services without a trusted transaction.

IV. DESIGN PARAMETERS

Communicating information such as operational state and available services requires high reliability, low-latency and computing resources. The information exchange mechanism, a platform for space assets to request and share valuable data and services on-orbit, must be resilient to a variety of space system attacks [37]. It also must be open so that new systems can engage with the marketplace without previous participation. Given the safety-critical nature of space systems, real-time operations must be feasible and not limited to space vehicles within the same constellation. The exponential growth of space vehicles requires the broadcast platform to be fungible and scalable as many more space vehicles will need to be added in the future. Finally, the integration for such systems must be seamless to encourage adoption of the platform. Any asset should be capable of joining, regardless of its resources - assuming connectivity is available.

Five governing interaction principles must be achieved for the ORBITS architecture to achieve a zero-trust services transaction. Nodes must be able to:

1) request applicable services with an associated contract;
2) fulfill service requests;
3) contest the validity of a received offer if they find it in violation of the specified policy;
4) facilitate a consensus across the blockchain to confirm transaction validity; and
5) manage a ''good standing'' system to punish bad actors.

Each principle is depicted in Figure 2. These principles are embodied in the below approach and associated architecture.

V. METHOD

Our architecture will help to facilitate a process to engage each transacting node in a zero-trust exchange. The proposed approach begins when satellite X requests a service to be fulfilled by another satellite on the shared network. To begin, satellite X must publish this request across the ledger. The request takes the form of 'blueprint': a set of instructions defining how the service will be fulfilled. All service-providing satellites will be regularly listening for requests, checking for new activity. Upon seeing a service request, satellite Y may decide that they wish to fulfill this request. Before proceeding, Y will first check the distributed ledger to determine if payload X is in *bad standing*. The bad standing status is designed to flag exploitative parties and encourage participation in the consensus (otherwise, payloads have no incentive to dedicate processing abilities to verifying transactions). Y only proceeds if they determine that this is not the case. Y then formulates a response and publishes it to the ledger in a partly encrypted form. This allows every satellite listening to the ledger to observe if the response fits the requests presented by Y without revealing potentially sensitive data in Y's response. X now has an opportunity to accept or reject Y's response. If X is satisfied, the exchange proceeds as per the request and the blueprint

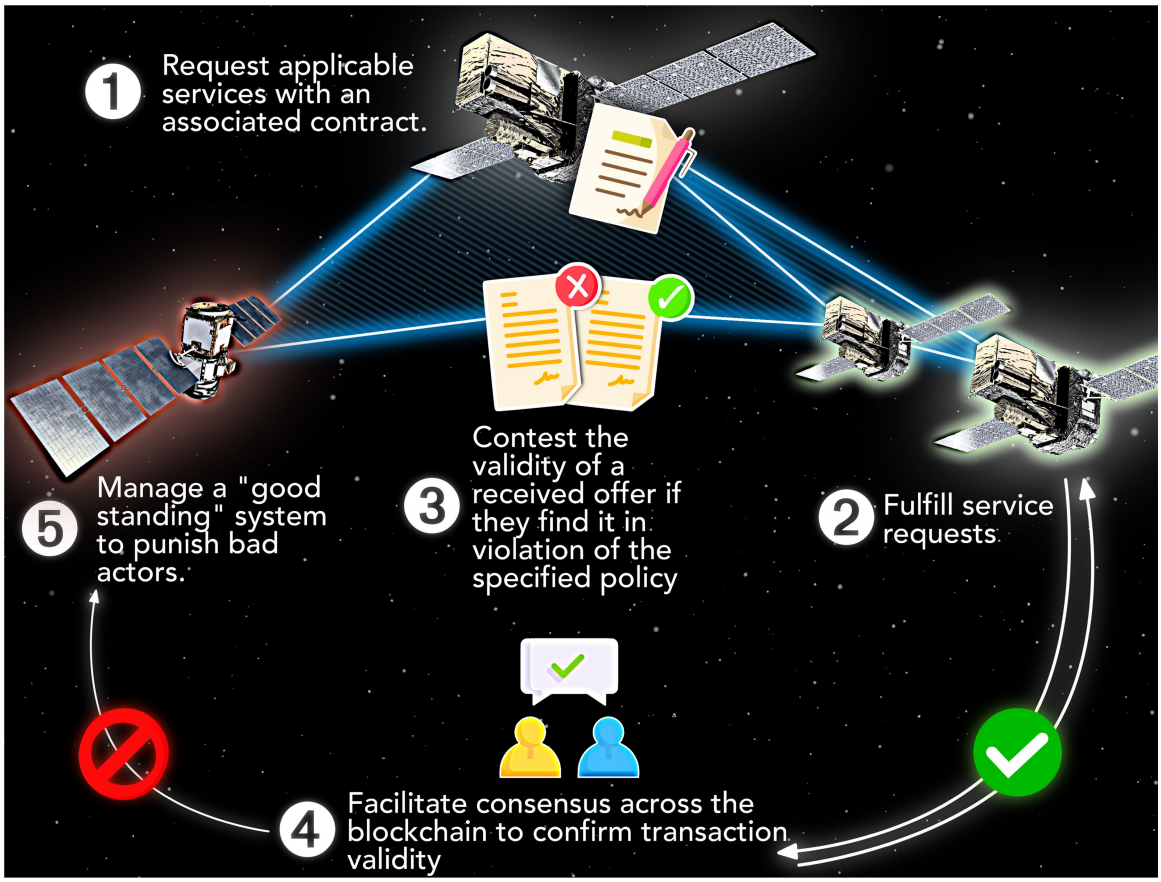

FIGURE 2. Governing principles for zero-trust service exchange.

does not need to be validated by others on the ledger, thereby not computationally taxing the network. Otherwise, if X contests the integrity of Y's service, X requests validation of the contract over the distributed ledger, thereby converting it to a smart contract. Every payload on the ledger has the opportunity to certify the integrity of Y's response to the contract to assess if it matched the requested blueprint. The node, either satellite X or Y, whom the consensus sides against is given a strike towards the bad standing status, which is documented by the smart contract validators and posted across the ledger. Strikes towards bad standing can also be given due to a lack or participation in the consensus, which will be monitored by the distributed service. Importantly, contract validation that requires computational resources of participants is only requested when there is a dispute. When there is no dispute, contracts are posted to the ledger without validation and later pruned.

A. ARCHITECTURE

The below architecture and associated steps will facilitate the process described. This process corresponds to the system diagram depicted in Figure 3.

1) ORBITS nodes must be able to request any given service. The requester should be able to define the specifications for a requested service by stipulating a policy template, or 'lock'. Meeting these specifications will predicate accepting a service offered thereby enabling the features of a software-defined network.
2) Nodes must be able to make offers to fulfill service requests. This offer, accompanied by a key, will have to be encrypted in such a way as to protect the integrity of the service's contents while remaining transparent enough for an outsider to determine if the key conforms to the defined policy specifications.
3) Upon receiving a fulfillment offer, a requester must have the ability to contest the validity of a given offer if they find it to be in violation of their policy.
4) In the event of a contested transaction, ORBITS must facilitate a consensus across the blockchain to determine whether the requester or the supplier is at fault, thus presenting the lock and key to be validated by all payloads across the ORBITS architecture.
5) ORBITS must be capable of tracking a node's 'good' or 'bad' standing to punish bad actors. Punishments must be given to nodes which have a consensus against them (could be the requestor or proposer), or who repeatedly fail to contribute to the consensus of others' transactions.

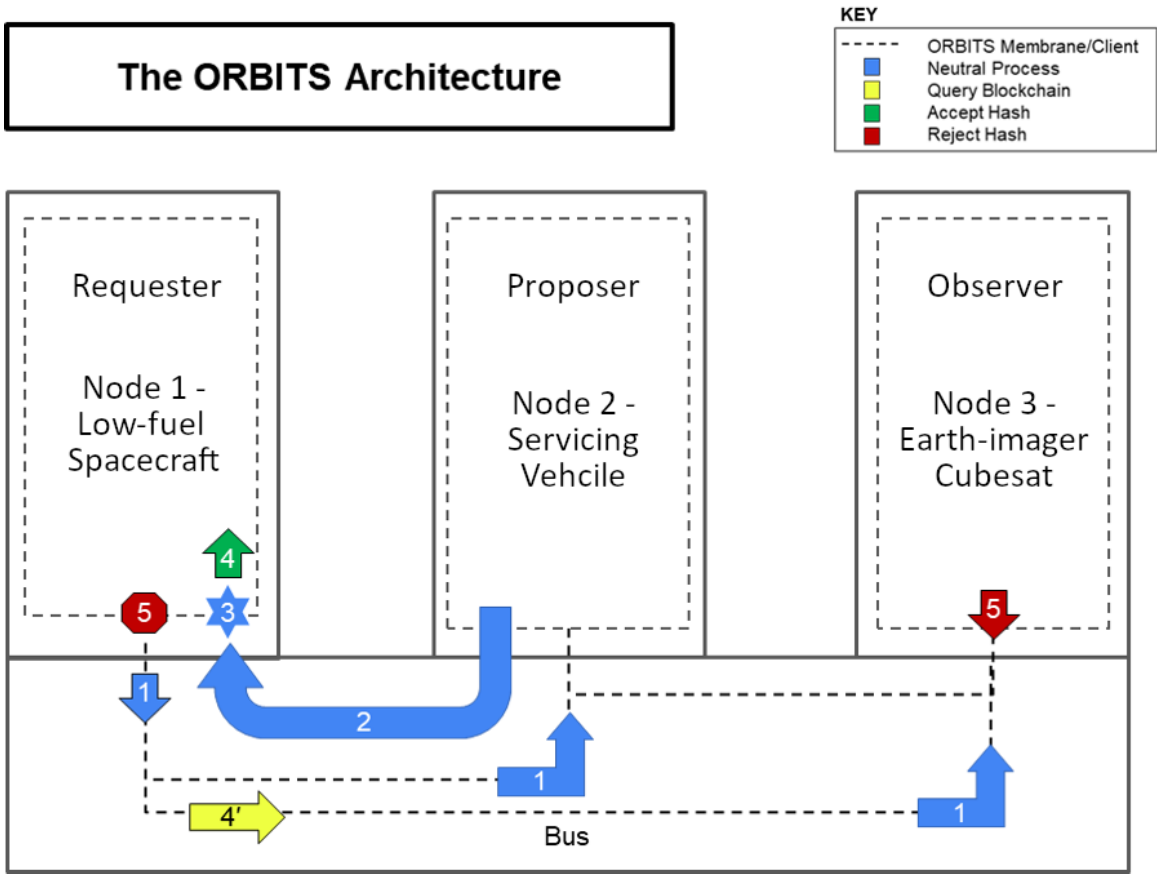

FIGURE 3. The ORBITS Architecture, demonstrating a scenario where a spacecraft running low on fuel requests the services of a refueling spacecraft. A cubesat in proximity is among the observers on the ORBITS network able to facilitate consensus if necessary.

Algorithm 1 ORBITS Service Transaction

0: ORBITS_Service_Genesis()
0: Broadcast Contract Request $\mathcal{R}$(Parameters x[], window t_r)
0: Proposals $\mathcal{P} \leftarrow$ []
0: **while** $t < t_r$ **do**
0: **if** Incoming proposal p satisfies x[] **then**
0: Append($\mathcal{P}$, p)
0: **end if**
0: **end while**
0: Select p' from $\mathcal{P}$
0: Mint block $B(\mathcal{R}, \mathcal{P})$ to DLT
0: **wait until** physical servicing complete
0: Internally validate service
0: **if** service disputed **then**
0: Flag B as disputed
0: **wait until** Consensus(B)

B. IMPLEMENTATION

In practice, we envision the ORBITS architecture as a software package that enables spacecraft and payload operators to gain access to a wider network of integrally shared services. By necessity, this network is built on the back of existing satellite networks. Figure 3 demonstrates how the network operates in the context of an ISAM service request. Note that the requester, proposer, and observer satellites (of which there would be multiple could be operated by three entirely unaffiliated entities. Algorithm 1 steps through the process of the genesis of a service request, solicitation of proposals, service execution, and the validation and consensus as necessary. These steps are outlined in more detail in the following material.

When a satellite joins the network as a node, it is initialized to a baseline standing, where it has access to the full privileges of the ORBITS architecture. If a consensus determination is made against an agent, it loses a significant amount of standing. Repeatedly loosing reputation this way can bring an agent to some minimum threshold, by which they are blocked from sending new contract requests. Fulfilling requests and participating in calls for consensus can gradually increase standing. As an incentive for constructive behavior, a node's standing is used to weight the priority queue for transmitting contracts across the network. Each node will store the consensus state of 'standing' for each other node in the network to avoid interactions with non-cooperative agents. Additionally, each node's ledger will store a record of service transactions that have occurred on the network. The use of ledger pruning algorithms to minimize impact

on resource-constrained payloads is under consideration [38]. Light-weight smart contract distributed ledger nodes similar to those previously demonstrated for use in autonomous vehicles will be considered [39].

Contract creation follows the format of the terrestrial Service Level Agreements (SLAs) that are used to enforce the terms of contracts. By encoding the quantitative terms of the transaction, the contract leaves an artifact of metadata usable in the validation phase. In the case of a refueling transaction, the contract would specify the time, location, type of fuel, amount of fuel, docking interface, and other details as applicable.

The validation step performed by orbits nodes upon contract receipt or consensus participation presents a challenging problem for implementation. Nodes performing validation should ideally have access to a maximal amount of metadata about the contract, its terms, and how it has been fulfilled without necessarily having received the contents of the service itself. We propose a physics-based validation procedure that analyzes the known intrinsic elements of a service exchange. For example, knowledge of a satellites' ephemera and the time of servicing could enable an agent to back-propagate if the servicing vehicle could have possible rendezvoused at the specified time, and knowledge of a satellite's weight before and after executing a refueling contract could provide insight to if the proper amount of fuel was transferred.

The ORBITS architecture does not propose a novel approach for key distribution supporting the distributed ledger. Distribution of standard has keys (e.g. SHA 256) from a trusted source would be sufficient to support this architecture. Should it become a feasible expectation for a diverse network of satellites, a Quantum Key Distribution solution would be suitable.

The ORBITS architecture has been conceived with the intention of inter-satellite operation. Expanding the network in this way entails the use of two potential systems: inter-satellite data links and ORBITS-enabled ground infrastructure. Of particular interest in these regard is the proposed Space-BACN relays, which would enable high-speed inter-satellite links and serve as a common link segment standard and companion to the ORBITS software.

VI. LIMITATIONS

Our vision and its requisite requirements presents a variety of technical challenges and risks. First, the consensus certification approach can be ineffective when there are few payloads that participate in the architecture. A bad actor can overtake 51% of the nodes, more easily, with fewer participants. Therefore, a critical mass of participants is necessary for the architecture to be effective. At first, we will set a to-be-determined minimum threshold number of participant nodes for ORBITS to operate. We may explore opportunities to leverage public blockchains in order to address constellations with few payloads that need to share services without the risk of ORBITS being overtaken.

Another apparent challenge is that ORBITS places increased computational strain on a class of already resource-constrained systems. The first consideration is that, ideally, the compute required by a payload to participate in ORBITS is less than the system would have otherwise required to run independently as a vertically integrated system. Additionally, measures will be taken to minimize the resource cost of using ORBITS. Parameters for ORBITS operations such as limiting validation to disputed transactions and stipulating the number of participating nodes required to achieve consensus will help reduce the resource draw of participating nodes.

Another concern is that an adversary could discover a means to abuse this architecture via a denial-of-service attack on an ORBITS ecosystem. We plan to extensively test a range of methods–including rate-limiters, strike systems, etc.–to determine the best ways to prevent such exploits.

Portability also presents a challenge, especially considering a hosted-payload environment where components are managed by a diverse set of stakeholders and could be running widely varying, often proprietary, operating systems and software suites. We plan to demonstrate our results on commercial off-the-shelf hardware in our lab and are open to engaging with industry partners to develop support for diverse, proprietary environments.

VII. DISCUSSION

We expect there will be several quantitative benefits to the ORBITS architecture compared to current mechanisms available to secure inter-agent interactions. First, we expect ORBITS to be resource efficient compared to other integral interaction services such as homomorphic encryption, which requires extensive power requirements not available on most space vehicles. The team's experience building light-weight, edge-based blockchain clients will be valuable here. Second, the fully distributed nature of ORBITS will reduce the net weight and cost that any given payload incurs to participate because each does not need to be vertically integrated. Finally, ORBITS will help to expand the lifespan of any satellite given its modularity. Should a service (e.g. a sensor) no longer perform as intended as supplied by one provider payload, a payload requestor could request the same sensor service from a different provider, without experiencing extensive downtime assuming the availability of the required services across participants in the ORBITS architecture.

We foresee several qualitative benefits of ORBITS. The ORBITS architecture will enable impromptu interactions between payloads without any prior configuration beyond installing the ORBITS client on the participant system. The ORBITS architecture can rapidly identify bad actors that are not complying with service request policies and block their further engagement. Additionally, should a single node be compromised, others across ORBITS will be unimpeded given its fully distributed nature. Integrity-preserved provenance is a core feature of permanent ledgers such as blockchains where 51% of the nodes would need to be compromised in order to disrupt inter-agent

consensus. Perhaps most significantly, we perceive the ORBITS architecture as introducing the possibility of new mission concepts of operation (CONOPS) via the proliferation of a distributed resilient network. This approach fosters more efficient mission design by eliminating the need for vertical integration in space systems: a system can delegate non-essential or infrequently-utilized services to other members of the network. Conversely, this also can spur development of small businesses to fill particular niches by adopting this burden. Furthermore, a platform that is mutually appealing to a wide variety of stakeholders introduces paths for cooperation that would otherwise be impossible – for example, an American asset contracting a Chinese asset to complete a data downlink would be possible within the vision for the ORBITS architecture.

VIII. CONCLUSION

It has been shown that while the need for interconnected satellite operations has never been greater, progression towards a system-agnostic service exchange marketplace in the domain has been stagnant. The deployment of the ORBITS architecture would constitute a revolutionary advancement for space systems because it will help incentivize the space sector to develop systems meant to scale and build on other's capabilities. A detailed analysis of the necessary behaviors of a service requester and provider demonstrates the possibility for service exchange in environments where nodes of the network may not necessarily have a trusted relationship. The fully distributed, provenance preserving and consensus-driven nature of ORBITS will facilitate a truly zero-trust mechanism for engagement with the benefits of a software-defined network, which has not been achieved previously for space systems. The ORBITS architecture could advance space system capabilities far beyond their expected use cases given it provides for versatility and agility to system services. These are qualities of missions that have been elusive thus far, thereby limiting the ability for the space sector to truly scale. The ORBITS architecture would also allow for more resilient constellations as it could help provide quick-turn recovery should a service fail.

ACKNOWLEDGMENT

Thanks to Greg Kuperman for his mentorship and support.

REFERENCES

[1] (2024). *Eutelsat Group*. [Online]. Available: https://oneweb.net/our-network
[2] (2024). *SpaceX*. [Online]. Available: https://www.starlink.com/
[3] (2024). *Defense Advanced Research Projects Agency (DARPA)*. [Online]. Available: https://www.darpa.mil/program/blackjack
[4] (2024). *Amazon*. [Online]. Available: https://www.aboutamazon.com/what-we-do/devices-services/project-kuiper
[5] Planet Labs. (2024). *Planet | Insights—Our Constellations*. [Online]. Available: https://www.planet.com/our-constellations/
[6] (2024). *Our Technology | Globalstar*. Accessed: Globalstar. [Online]. Available: https://www.globalstar.com/en-us/about/our-technology
[7] F. A. Fernández, "Inter-satellite ranging and inter-satellite communication links for enhancing GNSS satellite broadcast navigation data," *Adv. Space Res.*, vol. 47, no. 5, pp. 786–801, Mar. 2011.
[8] Viasat. (2024). *Intersatellite Communications*. [Online]. Available: https://www.viasat.com/space-innovation/space-systems/intersatellite-communications/
[9] H. Kaushal and G. Kaddoum, "Optical communication in space: Challenges and mitigation techniques," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 57–96, 1st Quart., 2017.
[10] O. Brown and P. Eremenko, "The value proposition for fractionated space architectures," *Space*, no. 7506, 2006.
[11] O. Brown, P. Eremenko, and P. Collopy, "Value-centric design methodologies for fractionated spacecraft: Progress summary from phase I of the DARPA system F6 program," in *Proc. AIAA SPACE Conf. Exposit.*, Sep. 2009, p. 6540.
[12] M. O'Neill, H. Yue, S. Nag, P. Grogan, and O. de Weck, "Comparing and optimizing the DARPA system F6 program value-centric design methodologies," in *Proc. AIAA SPACE Conf. Expo.*, Aug. 2010, p. 8828.
[13] M. Mosleh, K. Dalili, and B. Heydari, "Optimal modularity for fractionated spacecraft: The case of system F6," *Proc. Comput. Sci.*, vol. 28, pp. 164–170, Jun. 2014.
[14] E. Ong, O. Brown, and M. J. Losinski, "System F6: Progress to date," in *Proc. 26th Annu. AIAA/USU Conf. Small Satell.*, vol. 12, no. 3, 2012.
[15] A. Golkar and I. Lluch i Cruz, "The federated satellite systems paradigm: Concept and business case evaluation," *Acta Astronautica*, vol. 111, pp. 230–248, Jun. 2015.
[16] P. D. Collopy and P. M. Hollingsworth, "Value-driven design," *J. Aircr.*, vol. 48, no. 3, pp. 749–759, 2011.
[17] D. Selva, A. Golkar, O. Korobova, I. L. I. Cruz, P. Collopy, and O. L. de Weck, "Distributed Earth satellite systems: What is needed to move forward?" *J. Aerosp. Inf. Syst.*, vol. 14, no. 8, pp. 412–438, Aug. 2017.
[18] H. Natarajan, S. Krause, and H. Gradstein, *Distributed Ledger Technology and Blockchain*. Washington, DC, USA: World Bank, 2017.
[19] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[20] M. Nesbitt. (2019). *Deep Chain Reorganization Detected on Ethereum Classic*. Accessed: Dec. 2, 2019. [Online]. Available: https://medium.com/the-coinbase-blog/ethereum-classic-etc-is-currently-being-51-attacked-33be13ce32de
[21] V. Buterin. (2016). *What is Ethereum?*. [Online]. Available: http://www.ethdocs.org/en/latest/introduction/what-is-ethereum.html
[22] I.-C. Lin and T.-C. Liao, "A survey of blockchain security issues and challenges," *IJ Netw. Secur.*, vol. 19, no. 5, pp. 653–659, 2017.
[23] E. Reilly, M. Maloney, M. Siegel, and G. Falco, "An IoT integrity-first communication protocol via an Ethereum blockchain light client," in *Proc. IEEE/ACM 1st Int. Workshop Softw. Eng. Res. Practices Internet Things*, May 2019, pp. 53–56.
[24] G. Falco, C. Li, P. Fedorov, C. Caldera, R. Arora, and K. Jackson, "NeuroMesh: IoT security enabled by a blockchain powered botnet vaccine," in *Proc. Int. Conf. Omni-Layer Intell. Syst.*, May 2019, pp. 1–6.
[25] K. K. Kondru and R. Saranya, "Directed acyclic graph-based distributed ledgers—An evolutionary perspective," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 1, Oct. 2019.
[26] C. Partridge, R. Walsh, M. Gillen, G. Lauer, J. Lowry, W. T. Strayer, D. Kong, D. Levin, J. Loyall, and M. Paulitsch, "A secure content network in space," in *Proc. 7th ACM Int. Workshop Challenged Netw.*, Aug. 2012, pp. 43–50.
[27] C. Liu and J. H. Anderson, "Supporting graph-based real-time applications in distributed systems," in *Proc. IEEE 17th Int. Conf. Embedded Real-Time Comput. Syst. Appl.*, vol. 1, Aug. 2011, pp. 143–152.
[28] J. d. La Beaujardiere, R. Mital, and R. Mital, "Blockchain application within a multi-sensor satellite architecture," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2019, pp. 5293–5296.
[29] R. Xu, Y. Chen, E. Blasch, and G. Chen, "Exploration of blockchain-enabled decentralized capability-based access control strategy for space situation awareness," *Opt. Eng.*, vol. 58, no. 4, p. 15, Feb. 2019.
[30] M. Feng and H. Xu, "MSNET-blockchain: A new framework for securing mobile satellite communication network," in *Proc. 16th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2019, pp. 1–9.
[31] Globe Newswire. (2020). *Xage Partners With the U.S. Space Force to Introduce Blockchain-Protected Zero Trust Cybersecurity for Space Architecture Resiliency*. [Online]. Available: https://www.globenewswire.com/en/news-release/2020/09/17/2095311/0/en/Xage-Partners-with-the-U-S-Space-Force-to-Introduce-Blockchain-Protected-Zero-Trust-Cybersecurity-for-Space-Architecture-Resiliency.html

[32] PR Newswire. (2019). *Spacechain Receives Support From European Space Agency for Blockchain Satellite Technology*. [Online]. Available: https://prnewswire.com/news-releases/spacechain-receives-support-from-european-space-agency-for-blockchain-satellite-technology-300920274.html
[33] J. Hill. (2022). *Spideroak Wins Dod Contract to Demo Orbitsecure Zero-Trust Protocol*. [Online]. Available: https://www.satellitetoday.com/government-military/2022/11/02/spideroak-wins-dod-contract-to-demo-orbitsecure-zero-trust-protocol/
[34] K. L. Jones, "Blockchain: Building consensus and trust across the space sector," in *Proc. 35th Space Symp. Tech. Track, Colorado*, 2019, vol. 4, no. 8, p. 19.
[35] A. M. Long, M. G. Richards, and D. E. Hastings, "On-orbit servicing: A new value proposition for satellite design and operation," *J. Spacecraft Rockets*, vol. 44, no. 4, pp. 964–976, Jul. 2007.
[36] N. G. Gordon and G. Falco, "Reference architectures for autonomous on-orbit servicing, assembly and manufacturing (OSAM) mission resilience," in *Proc. IEEE Int. Conf. Assured Autonomy (ICAA)*, Mar. 2022, pp. 124–128.
[37] G. Falco, "The vacuum of space cyber security," in *Proc. AIAA SPACE Astronaut. Forum Expo.*, Sep. 2018, p. 5275.
[38] V. Buterin, "State tree pruning," *Ethereum Blog*, vol. 26, Jun. 2015. Accessed: May 20, 2024. [Online]. Available: https://blog.ethereum.org/2015/06/26/state-tree-pruning
[39] G. Falco, G. Falco, J. E. Siegel, and J. E. Siegel, "A distributed 'black box' audit trail design specification for connected and automated vehicle data and software assurance," *SAE Int. J. Transp. Cybersecurity Privacy*, vol. 3, no. 2, pp. 97–111, Oct. 2020.

GREGORY FALCO (Member, IEEE) is currently an Assistant Professor with the Sibley School of Mechanical and Aerospace Engineering, Cornell University. He is the Director of the Aerospace Adversary Laboratory, that focuses on developing secure and autonomous space systems. He has been recognized as a DARPA RISER. He was a recipient of the DARPA Young Faculty Award and was recognized in Forbes 30 Under 30. He is the Chair of the IEEE Standards Association's International Technical Standard for Space System Cybersecurity.

NATHANIEL G. GORDON received the master's degree in systems engineering from Johns Hopkins University. He is currently pursuing the Ph.D. degree in aerospace engineering with the Sibley School of Mechanical and Aerospace Engineering, Cornell University. His research interest includes developing mission-resilient space technology. He was a recipient of the Johns Hopkins University Applied Physics Laboratory's Graduate Student Research Fellowship from the Johns Hopkins University Applied Physics Laboratory.