A Hybrid Architecture for Secure Management of Manufacturing Data in Industry 4.0

Anku Adhikari
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA
Email: aadhikr2@illinois.edu

Marianne Winslett
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA
Email: winslett@illinois.edu

***Abstract*—In this paper, we analyze the suitability of the methods available today for securely managing the wide variety of data produced by the manufacturing sector. We propose a hybrid information architecture for manufacturing, based on decentralized blockchains, cloud-based WORM storage and ordinary cloud storage. We point out shortcomings in the technology available today for realizing this architecture. In particular, we identify a need for low-cost IoT-based systems to capture, identify, preprocess, encrypt and transmit factory floor data to the corresponding data storage subsystems. We describe our proof-of-concept implementation of such an IoT system, along with the factory case study that inspired it, and argue that this system is sufficiently inexpensive to be retrofitted into today's factories.**

***Index Terms*—Data Architecture, Manufacturing, Security, Digital Threads, Big Data, IoT, Blockchain, WORM**

I. Introduction

Opportunities in today's global market increase competitive pressures and push companies toward cooperation and flexibility. To adapt, manufacturers are moving toward Industry 4.0, which features new concepts enabled by technology:

1) Interoperability: Machines, devices, sensors, people will be connected and communicate, largely via IoT devices.
2) Information transparency: A virtual copy of the world will be collected through sensor data.
3) Technical assistance: Information will be aggregated and visualized in a human-friendly form to assist in decision-making. Further, cyber-physical systems will carry out tasks that are not practical for humans.
4) Decentralized decisions: Cyber-physical systems will make decisions for themselves, becoming as autonomous as possible.

Extreme examples of Industry 4.0 include Amazon's robot-run warehouses [1] and the "lights out" factories of FANUC and Philips [2].

Manufacturing data plays a crucial role in Industry 4.0 by providing a digital snapshot of each activity: design and process details, provenance of subparts and raw materials, and quality assurance information.

Data Capture. On and off the factory floor, modern heavy machinery often has built-in sensors that monitor the internal state of the machine and send their data to an associated local or cloud-based PC for analysis. Heavy machinery may last for decades, so its initial sensing and computing capabilities eventually become antiquated. However, factories can easily attach new sensors to these old machines and transmit the data to a new PC.

Analytics. Deployed manufacturing use cases for captured data range from factory floor level asset lifecycle tracking to supply chain and infrastructure management [3]. More advanced analytics can enable better coordination, help improve manufacturing cost effectiveness and support just-in-time manufacturing. For example, Airbus is exploring provenance data tracking for plane parts [4]. Everledger and De Beers are experimenting with blockchains to keep blood diamonds [5] out of the jewelry manufacturing supply chain. Rolls-Royce uses sensor data analytics to extensively monitor the health of deployed machinery [6].

Digital Threads. All the information described above can be viewed as part of the *digital thread* for a high-value manufactured item such as a car or airplane. The thread may be contained within an enterprise or stretch across many stakeholders who supply subparts, provide certification and testing services, or handle outsourced steps in the manufacturing process. A digital thread can provide indisputable accountability information linking factories to their products, but requires securely capturing thread data, transporting it to a suitable location, and applying appropriate access rights for contractually-required sharing.

Affordability. Although the required sensors themselves are usually inexpensive, the need for on-site staff with computer expertise to extract useful information from data tends to limit the adoption of high-tech approaches by most manufacturers. Over time we expect that cloud-based analytics packages will become available that allow this technology to permeate the long tail of small and medium size factories that constitute the vast majority of manufacturers. The real-time cloud-based analytics based on sensors on heavy machinery sold by Caterpillar and John Deere have already enabled a similar transformation for US farmers, providing detailed situational awareness of their fields and enabling fine-grained customized treatment of each square meter of cropland.

Contributions. In this paper, we propose a hybrid data management architecture based on the threats to the different types of information on the factory floor and when, how,

978-1-5386-9151-9/19/$31.00 ©2019 IEEE

and by whom the information is to be used. To focus the discussion, we rely on a case study of a real factory. We also describe the design and implementation of a low-cost factory floor information capture system that addresses key threats and can be retrofitted into factories as they move toward Industry 4.0.

The remainder of the paper is organized as follows: Section II surveys related work. Section III presents the factory case study and hybrid architecture. Section IV presents our proof-of-concept implementation and Section V concludes the paper.

II. Background and Related Work

The Linux Foundation's open-source Hyperledger data management framework for manufacturing is based on blockchain technology [7]. Hyperledger's contributors aim to support more efficient industrial collaborations by providing standardized tools and platforms for enabling data integrity, validation, and auditability through blockchains. Hyperledger protocols focus on use cases in supply chain data sharing and management [8], where multiple independent stakeholders collaborate in a trustless network to transport a product from one point to another.

The major advantage of blockchains for digital threads is their truly distributed nature and native consensus support. Blockchains provide data immutability, provenance, consistency, and failure tolerance, and are verifiable and data auditable using built-in cryptographic mechanisms. These features are all a good match for the low-trust environment of a manufacturing federation. Combining blockchains with smart contracts will enable automated enforcement of some conditions in real-life contracts. We see major companies standardizing blockchain methods and requiring suppliers to join their consortia. For example, Walmart and Airbus are both working with Hyperledger.

Shared supply chain data and its threat model are a good match for blockchains, but shared supply chain data is only a small portion of a digital thread. Much other fine-grained data from a factory floor can be valuable for manufacturing analytics, but is a poor match for blockchains, due to its volume, velocity, and mismatch with the blockchain threat model, as discussed below.

Confidentiality is a major concern for manufacturers. Data generated at the factory floor can reveal sensitive manufacturing intellectual property that could compromise a manufacturer's competitive advantage, such as manufacturing processes, design strategies, and customer and supplier lists. Blockchains do not inherently support confidentiality, so blockchain entries must employ suitable cryptographic methods to protect the confidentiality of the data appended, while still supporting verification of entries.

To this end, some recent blockchain protocols have used homomorphic encryption (e.g., Elements Alpha), zero-knowledge proofs (e.g., Hawk) or hierarchical deterministic wallets to improved data and user confidentiality. Such techniques provide confidentiality while still allowing verification, but they are too computationally expensive to serve as a general solution for the confidentiality issues associated with data sharing in manufacturing. Encryption of data would impose a significant burden on IoT-enabled small factory-floor machines.

Scalability of most blockchain protocols depends on their block size and clock frequency, and this would pose a challenge if the manufacturing activity required a large amount of data to be shared and block computations to be done frequently [9], [10]. More generally, wide-scale IoT-level deployment is impractical. Throughput and latency have been improved by advances such as BitCoin-NG, yet major bottlenecks lie in the network diameter size and node processing power [11]. Limiting the network diameter translates to limiting the number of machines able to mine and contribute to a blockchain on the factory floor. If fewer miners are present, the data gets distributed accordingly and can overload the miners unless they are quite powerful, which raises affordability issues for smaller factories.

At the other end of the spectrum, traditional file systems and databases scale up and out extremely well. When needed, they can provide fine-grained control over data access through their front ends. However, they are under the control of their owners, which makes them vulnerable to insider attacks that tamper with data. For example, a company that sells a faulty component that is later found to be responsible for a significant failure, the company will have an incentive to delete the quality assurance data that shows it to be at fault. If a hash or other fingerprint of the data has already been recorded on a public blockchain, however, the company can be presumed guilty if it does not provide the data corresponding to that fingerprint.

Traditional databases are also vulnerable to sabotage from within, e.g., from disgruntled employees, both en route and once it is stored. Human error can also easily lead to the removal of data that is contractually obligated to be retained for extended periods. While the use of public clouds removes the potential for insider attacks that require physical access to the data, any superuser can still tamper with the data, maliciously or inadvertently.

These problems can be reduced by the use of WORM storage for data that must be retained long-term. WORM storage can be thought of as a file system that understands mandatory retention periods and allows each file to be written only once. At the end of its retention period, a file can be deleted but not otherwise altered. WORM storage is very inexpensive and available from all major storage vendors; it is widely used for regulatory compliance, such as Sarbanes-Oxley. Because of the large volume of historical data that typically must be retained for long periods and will be rarely if ever read, WORM storage focuses on making writes efficient. An insider with physical access and sufficient expertise and privileges can tamper with WORM storage, but this risk can essentially be removed by using the WORM facilities offered by public clouds, e.g., by Amazon[12].

While researchers have investigated ways to leverage WORM storage to provide tamper-evident indexes for data lookup and even tamper-evident relational databases, publicly

available cloud WORM facilities are still just file systems. As brute force search is impractical, there is no trustworthy way to find a particular piece of information on WORM storage based on part of its content. In other words, WORM storage alone does not guarantee availability, because it does not support fine-grained content-based lookup.

Further, WORM storage is inherently centralized, which prevents its easy adoption in a manufacturing federation when information must be shared.

Previous work has also summarized the technical challenges specific to managing and securing digital threads [13]. For example, the data capture and management mechanism must be fast and tightly coupled to the activity that generates the data, as otherwise it could add lag time to the manufacturing process, increase the factory down time, and reduce the factory yield. As fine-grained real-time data from real-time processes on the factory floor can reveal manufacturing intellectual property secrets, the newly generated information must be kept safe from prying eyes while it is being transmitted from the factory floor to its long-term home, e.g., by using a secure channel or encrypting the data before transmission.

III. Case Study & Data Architecture

A. Datum Tool & Manufacturing

Our case study is based on Datum Tool & Manufacturing in South Elgin, Illinois (datumtoolandmanufacturing.com). This 33-year-old company is a full service machine shop that specializes in precision machining of specialty parts such as valves, end caps, and nozzles. Like other small US family-owned machine shops, the company has automated tools and computational support but no data collection or analytics support beyond what is built into individual factory floor machines. We refer to the company today as Datum 3.0.

Through factory visits and interactions with the owner and employees, we gained a high level understanding of Datum 3.0's data flows and processes. Figure 1 shows several internal process stations at Datum 3.0, as well as processes they outsource. In the figure, raw materials are procured and stored until needed for an order (1). Then they go through an iterative process of material removal that includes cutting to length, turning on a lathe and milling (both computer-controlled), grinding, and finally polishing (2-5, 7), interspersed with manual measurement and comparison to tolerances. The next step is heat treatment and plating, which are outsourced to another company (6). In-progress orders are tracked and moved between stations manually (8). Protective waxing (9) is the final step before packing and shipping (10); the product is also shipped to and from the outsourced plating service. Datum 3.0 has recurrent similar orders from key customers, and most orders come from regular customers.

When Industry 4.0 technology becomes cost-effective for small manufacturers, we can imagine automated movement of material between process stations, automated measurement, and automated tracking of the progress of orders. On the IoT side, sensors on factory floor machines could use audio and vibration data to detect when bits and other cutting and grinding edges are getting dull, before a human can recognize the change in sound or a low-quality pass through the machine has been made. Audio and vibration could also be used to detect machine failures before they occur, so that, for example, predictive maintenance could allow an out-of-calibration problem to be fixed before it causes down time and delays order delivery.

B. Data Features and Use Cases

The best way to handle a particular kind of data generated at a factory will depend on its size, arrival rate, threat model, how often it will be read and by whom, its anticipated lifetime, the information technology capabilities and infrastructure of the factory, and the cost and benefits of outsourcing data storage and/or analytics. The discussion that follows is based on six use cases for Datum 4.0, which we characterize in terms of these attributes:

- UC1: *Microphones and/or accelerometers near moving machine parts, for predictive maintenance analytics based on audio and/or vibration.* Very high volume and velocity; short read-once lifetime unless being used to train a new predictive maintenance model; internal access; as a side channel, can leak manufacturing IP, design IP, factory activities.
- UC2: *Thermometer senses 3D printer bed temperature, to support the quality assurance certification that the product was processed without exceeding a certain temperature.* Low to medium volume and velocity; very long read-once lifetime if contractually required for forensic purposes; internal or external access; side-channel leak of proprietary information, potential motivation to tamper.
- UC3: *Minute angle readings from a CNC mill as it grinds through raw metal to shape a bolt, for correlation with usage data about bolt lifetimes and eventual analytics for process improvement.* Very high volume and velocity; very long read-once lifetime; internal access; side-channel leak of proprietary information.
- UC4: *Tagging of raw material source and composition as part of a complete digital thread verifying source and supporting certification requirements to satisfy contractual terms of a many-part product manufactured by a federation.* Low volume and velocity; very long read-multiple lifetime; internal and external access within federation; incentives for tampering and theft (supplier information can be very attractive to competitors).
- UC5: *Product measurements for/from outsourcing,* as in the outsourced chrome treatment step (8) in Figure 1, to help both parties verify that the correct thickness of chrome was applied. Low volume and velocity, short read-rarely lifetime, access by two parties; incentives for tampering and perhaps for theft.
- UC6: *Product shipment tracking, for verifiable delivery.* Low volume and velocity, long read-rarely lifetime, access by three or more parties, modest incentives for tampering and theft (reveals customer names to competitors).

Fig. 1. Snapshots of a factory: Datum Tools & Manufacturing

The distinction between data that is only used internally within a manufacturer and data that is shared with outsiders is fundamental for storage and handling. The latter is made available to other members of the manufacturing federation as contractually or legally required, such as logistics companies, suppliers of materials and services, customers, customers' customers, and so forth. Many other key data characteristics tend to divide along the internal/external visibility line.

In general, tampering is a low-level threat for data that is only used internally; when such tampering occurs, it is most likely attributable to human error or a malicious insider. Data that is only used internally often can reveal IP, so theft is a concern. Internal-only data includes much high-volume, high-velocity data that can be fed into analytics models (read-once) or used to construct them (read-multiple). Typical internal-only data includes fine-grained sensor data used for predictive maintenance and process improvement, as in UC1 and UC3; such data provides a fine-grained snapshot of sensitive factory floor processes at each moment.

Internal-only data faces minimal risk of tampering or repudiation. This suggests that simple, readily available approaches to ensure confidentiality and integrity while also providing access for analytics are best. The most inexpensive approach is to consume the data immediately on a well-secured machine in immediate proximity to the factory floor, and then discard the data, but this is impractical since most manufacturers do not patch or update their computer software: patches often lead to factory downtime, and no patches are available for old heavy machinery software anyway. Instead, confidentiality will be better preserved by using cloud-based storage and services for internal-use data, even though a large volume of data must be transmitted to the cloud using a secure channel.

Use cases UC2 and UC4-6 involve externally-visible data, i.e., information that the factory has agreed to share with its collaborators. In all four of these use cases, there is a significant incentive for tampering to avoid blame when things go wrong.

For use cases UC4, UC5 and UC6, since the data is likely to be of low volume and velocity, blockchain based data management methods can be considered. For example, suppose Datum receives a shipment of steel bars under UC4. Datum can record this fact and the provenance of the shipment on a blockchain. If the shipper and supplier have entered their own information about the shipment on the same or other blockchains, then Datum can verify those entries and reference them in its own entry.

In UC6, the product shipment tracking data volume from any single manufacturer will be low, but the aggregate volume across all customers of a shipper may be very high. Given that the threat levels are relatively low, a blockchain-based approach seems too expensive. We prefer to see a system like that provided by shippers today, where anyone possessing an appropriate token can see the logistics details for the order associated with that token. These systems are based on traditional database management technology. If data integrity and availability are of high concern, the shipper could charge an additional small fee to guarantee data availability for a fixed period of time, e.g., by archiving the records on WORM storage for a fixed period of time.

For UC5, in theory Datum can record on a blockchain the dimensions of the product being sent for plating. In theory, the plating company can do the same when it completes the order and ships it back; and a smart contract can automate payment when the shipment is received. However, the recipient of shipped goods may dispute that the posted measurements actually correspond to the shipment received, and the blockchain cannot resolve this problem.

Further, if the order is large then the volume of data may be

too high to be practical for a blockchain. Also, Datum may not wish to make its product's dimensional data visible to others. As discussed earlier, the computer security community has a bag of tricks to preserve confidentiality in such a situation, but they are unlikely to provide a cost-effective solution to this problem. We return to these points in the next use case.

UC2 combines high volume and velocity with integrity and availability concerns for temperature data. Putting the data on the blockchain would address the latter two concerns, but is impractical for scalability reasons. We propose a hybrid solution: the data itself can go on low-cost cloud-based WORM storage owned by the manufacturer, while a token corresponding to the data can be included on the blockchain. The token leaks minimal information about the manufacturing process, allaying potential confidentiality concerns. Upon conditions that can also be spelled out in the blockchain, certain parties have the contractual right to present the token to the manufacturer and be given access to the data.

C. Hybrid Data Architecture Design

The Datum use cases suggest a three-tiered data architecture:

1) Traditional cloud-based storage and analytics services for high-volume, short-lifetime data that is not shared with others. This includes the vast majority of factory sensor data, which is at low risk for tampering but may be an appealing target for theft.
2) Blockchains for low-volume long-lifetime shared data. This data may be at risk for tampering, but theft is less of a concern.
3) A combination of cloud-based WORM storage and blockchains for high-volume shared data. Here the blockchain contains a token and conditions on when the token can be used. When the conditions are fulfilled, presentation of the token to the manufacturer grants the presenter with access to a particular data item that resides on WORM storage.

Figure 2 depicts the WORM-blockchain tier of the system. The WORM layer includes a blockchain element because even if data has been placed on WORM storage, there is no guarantee that anyone can find it. The WORM-blockchain combination overcomes this problem by placing the relevant lookup information on the blockchain.

For example, DELL EMC sells a WORM storage system where each stored file can only be retrieved by presenting a hash of its contents. If this hash and appropriate metadata, e.g., a part ID and manufacturer name, are placed on a public blockchain, then anyone who knows the part ID can, in theory, find the part ID on the blockchain and use the associated hash to retrieve the file about that part from the supplier's WORM storage in the cloud. The use of the blockchain ensures that the required token cannot be lost or tampered with, and reduces the availability problem to the need to quickly find a blockchain record by its content, which is already on researchers' agendas.

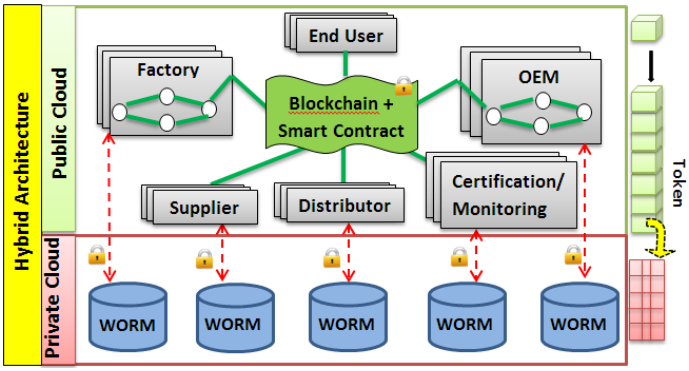

Fig. 2. WORM-Blockchain Hybrid Storage Tier

Hybrid WORM-blockchain storage may not offer the desired confidentiality and availability if a manufacturer divests a division, is acquired, or shuts down; if the WORM cloud provider shuts down; or the relevant blockchain is abandoned. These are interesting issues for future work.

IV. Proof of Concept Implementation & Challenges

A. Implementation of Datum 4.0 Data Capture & Storage

We used IoT-enabled compute units to build a low cost factory sensor data capture system for retrofitting factories. Our implementation provides separate data paths for data intended for internal use only (predictive maintenance, process monitoring and improvement) and data to be shared with external partners. We implement these as two separate subsystems supported by dedicated IoT compute units (see Figure 3). For internal-use-only data we chose the Particle Photon board as a compute unit, due to the small footprint that makes it easy to retrofit to legacy equipment, built-in encryption support, and good cloud integration and testing options. We integrated the Particle Photon board with a temperature sensor (TMP 36), microphone (Electret Microphone Amplifier MAX9814), and an IMU sensor unit (MinIMU-9 v5 with a gyroscope, accelerometer, and compass). The entire system cost US$50.40. We used AES and RSA to encrypt the data from the sensors before sending it to an Amazon Web Services cloud for storage.

Our implementation of the data capture system for externally shared data runs the Ethereum blockchain protocol on a US$40 Raspberry Pi 3B+ connected to several sensors ranging from $5 to $17. Our implementation includes two Raspberry Pi 3B+ boards that collect data from five (virtual) pieces of Datum factory floor equipment: the CNC mill, steel cutting, the raw material area, a factory-floor computer, and shipping. The testbed also includes a laptop running three blockchain miners for Datum, and a second laptop to support the external chrome plating service. When raw material arrives, the incoming smart contract goes to the raw material miner. When the material is processed at a piece of equipment, additional information is collected either manually or by automated measurements from the sensors/actuators connected to the Raspberry Pi unit,

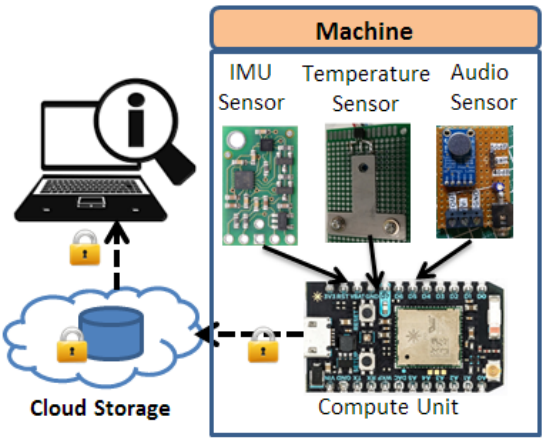

Fig. 3. Capture and Storage of IoT Data for Internal Use Only

and appended to the blockchain. Smart contract conditions can be set up in the server and the miner can fetch and check conditions through it. All pieces of equipment must be connected to a miner.

Support for integrating blockchains with the Photon Particle board was lacking, so it would have been difficult to use the Photon for externally shared data.

The Photon Particle board has built-in support for the Device cloud. As we wanted to evaluate its compatibility with other clouds, we had to decouple existing links with the Device cloud and link it with a more popular AWS cloud support instead. We observed issues with the stability of the Photon board connection in general and had to manually reset the connection if the device was idle for long. This could be remedied in a production system by implementing automated reset support.

Today's small-footprint IoT compute units are resource constrained. A Raspberry Pi's processor is considered fairly powerful for a small computer, but its I/O throughput is limited. In a Raspberry Pi and most other IoT-capable compute units, the Ethernet and all USB data go through a single USB 2.0/3.0 pipe, placing a hard limit on the volume of data coming into the system and going out to the blockchain. The throughput for the latest model of Raspberry Pi 3 is theoretically estimated to be 40Mbits/sec by Wifi or 300Mbits/sec by gigabit LAN. Memory and processing speed constraints may also dictate that the blockchain block sizes processed remain small, so that they do not delay the manufacturing process. In particular, the need to support cryptographic protocols for confidentiality requires both the blockchain mechanism and cryptographic protocols to be low-cost. The overhead of handling multiple sensors and preprocessing their data places additional burdens on the system. Overall, these limits further emphasize the need for a WORM-blockchain tier of storage if low-cost deployment is desirable.

V. Conclusions

The diversity of manufacturing data and applications means that no single data management solution available today will meet all performance, security, and cost requirements. In this paper we proposed a hybrid data architecture based on how different types of manufacturing data are stored, used, and shared, with an eye to the needs of the long tail of small and medium manufacturing enterprises. The resulting three tier architecture includes a tier of conventional cloud-based storage and services for high-volume, high-velocity, short-lifetime, sensitive data typified by sensor data from the factory floor. A second tier uses blockchains for low-volume data that must be shared with others, has a long lifetime, and may have high incentives for tampering. The third tier is for high-volume long-lifetime sensitive data that is sensitive, must be shared, and may have high incentives for tampering. This tier employs cloud WORM storage for the data itself, plus a blockchain entry that includes a token allowing retrieval of information from the WORM if certain specified conditions are met. Our PoC design and implementation focused on practical, cost-effective data capture systems for all three tiers, that can be easily retrofitted to today's factories.

References

[1] P. Tracy. Case study: Amazon embraces shipping automation, robotics. Last Accessed: 2018-11-01. [Online]. Available: https://www.rcrwireless.com/20160708/internet-of-things/amazon-automation-tag31-tag99

[2] J. Schweder. Turning out the lights on the factory floor. Last Accessed: 2018-11-01. [Online]. Available: https://www.automationworld.com/article/technologies/robotics/turning-out-lights-factory-floor

[3] K. Lewis. (May 2017) Blockchain: Four use cases transforming business. Last access 2017-10-17. [Online]. Available: https://www.ibm.com/blogs/internet-of-things/iot-blockchain-use-cases

[4] J. Holl. Blockchain: The trust protocol. Last Accessed: 2018-11-01. [Online]. Available: https://www.airbus.com/newsroom/news/en/2017/03/Blockchain.html

[5] L. Kemp. Putting bling on the blockchain: The everledger story. Last Accessed: 2018-11-01. [Online]. Available: http://institute.swissre.com/research/library/Rdm_Blockchain_Leanne_Kemp.html

[6] I. Bellamy, Woodrow. Oems embrace new aircraft engine health monitoring tech. Last Accessed: 2018-11-01. [Online]. Available: https://www.aviationtoday.com/2017/02/15/oems-embrace-new-aircraft-engine-health-monitoring-tech/

[7] Hyperledger, "Hyperledger whitepaper," last accessed 2017-10-17. [Online]. Available: http://www.the-blockchain.com/docs/HyperledgerWhitepaper.pdf

[8] J. Sevareid, *Use Case Inventory*, Hyperledger Requirements Working Group (WG), 2017, last updated 2016-11-01. [Online]. Available: https://wiki.hyperledger.org/groups/requirements/use-case-inventory

[9] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

[10] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. bft replication," in *International Workshop on Open Problems in Network Security*. Springer, 2015, pp. 112–125.

[11] I. Eyal, A. E. Gencer, E. G. Sirer, and R. V. Renesse, "Bitcoin-ng: A scalable blockchain protocol," in *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. Santa Clara, CA: USENIX Association, 2016, pp. 45–59. [Online]. Available: https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/eyal

[12] J. Barr. Create write-once-read-many archive storage with amazon glacier. Last updated 2016-11-10. [Online]. Available: https://aws.amazon.com/blogs/aws/glacier-vault-lock/

[13] A. Adhikari, A. Hojjati, J. Shen, J. Hsu, W. King, and M. Winslett, "Trust issues for big data about high-value manufactured parts," 2016.