Collaborative and Accountable Hardware Governance using Blockchain

(Vision Paper)

Ashish Kundu, Zehra Sura, Upendra Sharma
IBM T J Watson Research Center, Yorktown Heights, New York, USA
{akundu, zsura, upendra.sharma}@us.ibm.com

***Abstract*—Hardware systems are complex and the complexity of such systems is increasing with the adoption of cloud computing, artificial intelligence, and with the advent of cryptocurrency and Internet-of-Things. The increase in complexity makes it harder to address failure, security compromises, and lack of performance in hardware systems. Further, such systems need to be customized and integrated by several parties and stakeholders in order to implement specific end-to-end applications.**

In this paper, we make two contributions towards managing this complexity: (1) we present a vision[1] on how hardware system development and governance should be carried out, and (2) we advocate applying a fresh approach to the problem of hardware development and verification, going from a black-box driven model to a white-box model in order to improve trust, reliability and efficiency in a dynamic, collaborative and accountable manner. We think that it is essential to employ automated validation of technical and business specification as well as to carry out analytics and machine learning on the internal states of hardware devices and systems in order to dynamically determine and/or predict faults, security issues, performance issues and address them in a collaborative manner. We lay out the desiderata, challenges and issues in accomplishing this. We also present a blockchain-based governance platform for hardware systems as part of our solution.

I. Introduction

We live in an age where our day-to-day lives are dependent on complex and interconnected computing technology for instance power utility grids, self-driving cars, or health monitoring devices that are dependent on a highly secure backend service in the cloud. Hardware devices are at the heart of all of these complex systems. The complexity of the hardware systems is continuing to increase in order to cater to the evolution in computing such as adoption of artificial intelligence, cryptocurrency, use of IoT devices, and edge computing.

Complex systems depend on multiple interconnected hardware parts working together, but the devices that form the system are often developed or controlled by different entities. For example, company A can assemble systems comprising compute chips from company B and memory chips from company C. These systems can be used to build a computational cloud cluster or datacenter by company D, which then leases it to company E to run their application. In another example, customers of a power utility can also be collaborators for power generation (e.g. by connecting their own solar panels on residential buildings to the power grid) and for power regulation (e.g. by connecting their own smart home automation devices like NEST thermostats to the utility's control system). Internet-of-Things (IoT), data centers and self-driving cars are examples of such large scale complex collaborative systems. These systems themselves are composed of complex devices that have a motherboard and hardware components which are built using several chips and smaller components developed by different vendors. These devices also face complex challenges to diagnose failure situations.

One common cause of device failures is *component failure* where a component behaves outside the expected norm defined in its technical specification; this can cause failure of the individual component/chip as well as failure of other components/chips and may even cause a systemic failure. In the example of self-driving cars, consider that one of the chips in the control system starts to generate more heat from time to time. Eventually, it may affect other systems or may even cause a catastrophic incident such as the car catching on fire. If we can determine the heat generation by each chip in its operational states, with the data being analyzed not only by the vendor who built the car, but also by the organization that developed the self-driving car platform and by regulatory organizations simultaneously, then we can more reliably determine the cause of such heat generation and act to fix the problem before it becomes catastrophic.

Another cause of hardware errors is malicious attacks. Hardware systems may be compromised by security and data breaches and may have trojans embedded in them [1]. This is often diagnosed by collecting event data and analyzing abnormal event-occurrence patterns. In essence, in collaborative systems (where collaboration is either on a single motherboard or across a large number of complex devices) it is necessary to determine the root cause of errors either in realtime or as part of aposteriori analysis in a secure trusted manner for ascribing responsibility and accountability to individual components or parties of the system. Such an analysis is often impossible to carry out because of lack of accurate, reliable and real-time data/information that is sufficiently detailed.

Collecting real-time data across large collaborative complex systems is challenging because of the scale of the system and

[1] This paper presents a vision for developing future systems based on the collective cross-disciplinary expertise of the authors.

978-1-5386-9502-9/18/$31.00 ©2018 IEEE
DOI 10.1109/CIC.2018.00026

the heterogeneity of the components involved. To address these problems we propose introducing a standardized hardware-supported monitoring mechanism.

Another challenge is that when there is failure of some components, it is necessary to provide a proof of malfunction of components to vendors. The commonly adopted solution is to have hardware devices send data to a centralized server to record events. Such a solution suffers from trust issues as all stakeholder parties do not take part in an equal capacity in it. Event monitoring based solutions cannot support immutable provenance and lineage of data that may be needed in executing business contracts, regulatory compliance and audit requirements. Satisfying such requirements necessitates maintenance of records of various critical events and states (at each granularity, i.e. from a single motherboard to datacenter) in an auditable fashion. To address this problem we propose a blockchain-based solution, that would naturally support the required capabilities.

In summary, we see the above mentioned problems as two connected problems. First lifecycle management and smart monitoring of hardware components and fine-granular data collection at the motherboard level in a ”white-box” manner. Second, a trustworthy collaborative framework to perform analysis of the collected data in order to provide root cause analysis and accountability. In this paper we present a vision that addresses these two problems and present a solution to realize the vision. We propose the novel notion of a ”blockchain chip” that uses blockchain and smart contracts in order to support the vision.

II. State of the Art

With IoT, big data and ubiquitous computing, the paradigm of open-source collaborative development of hardware systems is gaining traction [2], [3], [4]. Standards and common interfaces for IoT device discovery and connectivity are being addressed by industry consortiums such as the Open Connectivity Foundation [4]. Open source code and use of standard library APIs are already pervasive in software development. The hardware community is also working to adopt similar models. For example, the OpenPOWER ecosystem[2] opens up the design and specification of the POWER architecture, enabling cross-industry adaptation, evolution and use of hardware designs and components. Similarly, RISC-V [5] is an open-source instruction set architecture that aims to pave the way for collaborative and extensible hardware platforms.

While the mechanisms are in place for widespread collaboration on hardware system development, such collaboration is not the norm today. Collaboration on hardware is more difficult than collaboration on software because hardware errors are relatively much more expensive to fix in terms of price, skill level required, and service disruptions or inconvenience to end users. Typically, computing hardware is built using motherboards that provision for multiple subsystems, e.g. processors, memory controllers, I/O controllers, interface connectors, etc. System integrators assemble computing systems using motherboards and multiple components/chips, possibly from different vendors. The performance of the overall system may depend on the proper functioning of multiple disparate chips. The system integrators rely on some combination of technical specifications, in-house testing, and business indemnity contracts when assembling systems using components from different vendors.

The system integrators are solely responsible for the working of the integrated device/system, and are accountable for a set of performance metrics that may include measuring time, reliability, availability, security, or power/energy consumption. Some of these measurements can be obtained using performance counters built into hardware. However, hardware counters are difficult to use in diagnosing system-level problems for a number of reasons. First, both the types of counters available and the usage mechanisms are very different for different components. Second, counters are not easily accessible; they may require digging deep into technical documentation or operating the device in a special mode. Third, it is hard to correlate counters across different components in the same system to get a coherent view of system-level operation. Fourth, it is inefficient, or sometimes not even possible, to adapt the granularity of measurement based on dynamic conditions, whereas effective troubleshooting requires information at the right level of detail, neither too fine-grained nor too coarse. As a result, performance issues in today’s complex systems are difficult to troubleshoot to determine the root cause.

Another critical aspect of hardware system development is security of such systems and how trustworthy they are. It is possible to embed hardware trojans in circuitry or even as another chip on a motherboard. Backdoors can be incorporated into a hardware design in order to support powerful adversarial actors. With the explosion of IoT devices as well as the increasing development of automated systems such as self-driving cars, such malicious capabilities need to be prevented from being added or they need to be detected in a timely manner.

Moreover, it is becoming easier to build a system by assembling commodity hardware components due to the steep decline in price of components, availability of free and open-source software, open standards, and easy access to information via the internet. As a result, hardware devices are increasingly being developed by entities that make use of modular architecture – they bring together third-party hardware building blocks and connect them to deliver devices such as phones, laptops, home-monitoring devices while assuming that such a composition will remain reliable, efficient, secure and trust-worthy. Often these entities are neither well-trusted nor skilled enough to appreciate why modular architecture often does not lead to ”compositional security” or ”compositional reliability” even though functional capabilities can be achieved. The assembled devices are not trustworthy due to the lack of rigor in security analysis, risk of compromise, and the possibility of backdoors and trojans incorporated in the hardware components. In such a scenario, collaborative governance of devices, tools and processes is essential to estimate the risk, to provide trust and to mitigate issues in a timely manner.

III. Our Vision

In the next decade or so, with the advent of large-scale IoT and data-centers, any failure, erroneous operation or faulty behavior can lead to multiplicative (if not exponential) impact on the trust, security, reliability, availability and performance of the overall system. In order to contain the harm from such impact, hardware will have to become more reliable, trustworthy, and capable of providing a root of trust to other components in the system.

A. Overview of the Vision

With the ease of assembling devices and ubiquitous digitization of services, it will be even more important for users of hardware devices to have the capability to monitor, analyze, and "see" what is going on inside a closed hardware device. Such capability needs to include support for automated verification and operational validation of the device against its technical specification. As a result, it should enable real-time, automated problem determination, root cause analysis, as well as detection of any trojans, malware and backdoors that may be present. Further, it should be possible to carry out real-time mitigation of identified issues.

Our vision is to enable an ecosystem that supports "white-box" testing, verification, problem determination, and security-state analysis for hardware devices. The ecosystem provides an accountable and auditable collaborative environment, which could be public, private or hybrid. Such an ecosystem is based on auditable systems that support transparency and provenance. These systems will include a storage service, analytics and AI capabilities, as well as a problem determination service. Such an ecosystem will aid in rapid design, development and integration of hardware systems. In addition, such an ecosystem will aid in testing and verification, since monitoring and sharing of internal states of hardware components is a first order function of our design.

A hardware device can be composed of multiple hardware chips and components developed by several vendors. Today it may not have the circuitry that can be tapped into for collecting operational and environmental data such as power consumption, voltages at different interconnects, temperature, data transfer rate, amount of data transferred, and so on. We envision that future chips and motherboards will be designed with interfaces to collect all pertinent data. However such data may be sensitive and may require privacy-preserving techniques to be applied before it is sent to external entities. Also, the data collected may be very voluminous (terabytes per hour or even more), and a local data filtering service will be needed to reduce the amount of data to be propagated. The functionality to monitor, collect, reduce, and privatize data can be implemented in a separate chip included on the motherboard. The chip will expose a well defined programming interface for easy and secure access to the data. Then the data can be analyzed locally or in a remote system in order to identify faulty chips, malfunctioning devices, and devices not meeting technical specifications or policies. Data received can be stored for auditability, provenance, and validation purposes, thereby making accountability possible in the ecosystem.

B. Desiderata for the Vision

In the context of such a vision, today's processes and technology for hardware system monitoring, management and governance are highly limited. In order to make the hardware systems trusted, efficient, reliable and compliant, the following desiderata need to be supported by the hardware system as well as by the distributed/de-centralized system that connects the local hardware capabilities together with analytics, machine learning and policies.

- Automation: Technical specifications of the hardware devices need to be automatically verified. Other requirements, such as business policies or regulatory constraints, should be verified and potentially enforced in an automated manner.
- Collaboration: Bringing in multiple stake-holders and parties in a collaborative manner to contribute to development and dynamic operation of large systems as well as tiny systems in large-scale deployment is essential. The stakeholders include chip vendors, members of organizations such as OpenPower and OpenConnect, system integrators, IoT developers, government regulators, system administrators, providers of services for monitoring and testing, and application developers.
- Transparency: Transparency is essential in the way members are collaborating. There needs to be transparency in the technical specification, the constraints and conditions of operation, the way testing and verification is carried out, the monitoring data collected and how they are analyzed.
- Privacy and Confidentiality: Data collected from the hardware devices during runtime should be anonymized to preserve privacy.
- Trust Assurance: The hardware systems need to be trustworthy, and this can be supported using cryptographic techniques as well as dynamic monitoring of the device behaviours.
- Auditability and Provenance: For regulatory requirements and trust assurance, the ecosystem should support auditability of the data, actions and events. For this purpose, provenance of the events across the system is essential.
- Data analytics and machine learning: Large-scale analytics on data and processes need to be supported by the system.

C. Challenges and Issues

Reliably collecting data of various components on a motherboard and across a large number of machines in a datacenter or sensor data coming from IoT paradigm poses the following challenges:

- Intra-system scale: On the motherboard of any reasonably complex hardware system, there are a large number of chips/sub-components from different vendors performing different functions at a very high clock speed (as per their spec). Monitoring these hardware components against their specification (either hardware specification or functional specification) is a difficult challenge.

- Inter-system scale: In a datacenter, there are a very large number of servers, network devices and IoT devices from heterogeneous vendors. Again, monitoring a data-center wide operation is a complex problem of scale.
- Scale of number of vendors and similar devices: There are a large number of vendors manufacturing hardware chips and components; each of them have similar but slightly different specifications, for instance bus clock and voltage of DDR3, SDRAM of differing denominations (2G, 4G, 8G), etc. Tracking the source of failures and ascribing accountability to vendors is a challenge.
- Standardization: Defining and using standards for specification and monitoring of various components across the complete stack, i.e. ranging from hardware components to software, is an important challenge.
- Dynamic monitoring: A dynamic monitoring system that is able to change the granularity of measurement of information as needed; which means a coarse level monitoring in normal situations but a fine grain monitoring in suspicious or problematic situations is needed.
- Data collection: A standardized, resilient and trustworthy system for collection and storage of monitored data across a large number of components. Standardization that helps the users to easily encode their technical specifications into monitoring rules is a difficult challenge. Making a reliable, scalable and trustworthy network to collect data is a known challenging problem.
- De-centralized problem determination: Analyzing real-time and past data to determine the root cause and pinpoint the problems that are occurring within one hardware device (or even within millions of hardware devices) is not easy and is perhaps undecidable. The challenge is how to develop a sufficiently efficient technology for this purpose.
- Machine learning and analytics: Classifying and learning from the data collected requires understanding the type, quality and details of the data being collected from a motherboard or a chip. With the ”black-box” philosophy of hardware development, the details about what data is being collected and what it means is unclear. We need standardization efforts across the hardware level and the machine learning and analytics level to achieve clarity in data-related formatting, reporting semantics, frequency of monitoring, and privacy and sensitivity requirements of the data. The type of problem analysis or monitoring we want to carry out and the kind of data we have shall determine the type of machine learning models to be developed (based on deep learning or otherwise). Prediction of whether one or more devices are going to fail is of immense importance in self-driving cars, in a data-center, and in IoT applications.
- Privacy and confidentiality protection: Data collected from within the hardware systems may contain highly sensitive information even though the intent is not to collect what is being computed, but more about the operational information and environmental states. Sometimes in order to debug the state, we may need to get access to the system-level data in real-time that may contain all types of data. Even for monitoring, the state information may be sensitive (e.g. encryption keys maybe revealed by how much energy is used and the operations carried out [6]). We need to ensure that privacy and confidentiality for the data is protected per the policy in real-time and access control for such data is appropriately managed.
- Closed system monitoring: In an autonomous system such as a SpaceX rocket [7], or spaceship, or a self-driving car, it maybe essential for the components to coordinate among themselves about the kind of data they need, the kind of ensemble of machine learning models to use, the problem they need to analyze, how much time they need to spend for mitigating a problem and what can be predicted about the states of the components and the overall system. No external entity must be involved in supporting such capabilities.

IV. Our Solution

In order to realize the vision described in the previous sections, we present a system design that includes:

1) a system-wide blockchain service that leverages blockchain technology [8], and
2) blockchain chips built into motherboards of hardware devices in the system.

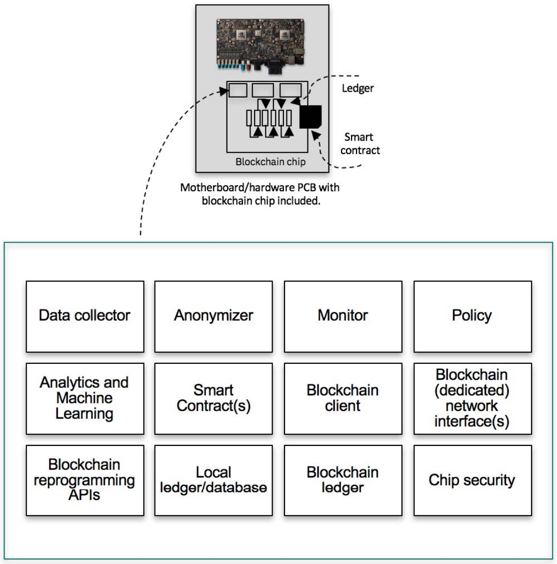
Ledger
Smart contract
Blockchain chip
Motherboard/hardware PCB with blockchain chip included.
Data collector
Anonymizer
Monitor
Policy
Analytics and Machine Learning
Smart Contract(s)
Blockchain client
Blockchain (dedicated) network interface(s)
Blockchain reprogramming APIs
Local ledger/database
Blockchain ledger
Chip security

Fig. 1. Blockchain Chip Architecture.

At the core of our system is the idea of placing a blockchain chip on a motherboard. A motherboard is the main printed circuit board (PCB) used to build hardware devices. It includes slots to hold the electronic components or chips that will comprise the hardware system, as well as communication links to allow these components to interface with one another. The blockchain chip (Figure 1) on each motherboard will monitor and report status, health, operational state, resource usage and/or performance of one or more components on the motherboard to a system-wide blockchain service. In particular, the data may include: timestamp, frequency of operation, power usage, temperature of the board, fan speed, performance and QoS numbers with min and max values, and other information that is accessible to the blockchain chip from any other given chip. By way of example, CPU temperatures can be monitored by reading the core temperature sensors of Intel, AMD, and other processors, and the sensors of ATI and NVidia video cards as well as SMART hard drive temperature can also be monitored.

Blockchain technology by design offers a trustworthy solution to information collection, exchange and analysis. Figure 2 illustrates the blockchain architecture, which has a blockchain network with multiple de-centralized (untrusted) peers. The peers share access to a fully replicated ledger, which serves as the trusted append-only system of record (SOR) and single source of transactional data. The blockchain network includes a consensus protocol, hash-chain, and the ability to evaluate smart contracts, all of which provide for trusted collaboration among the peers.

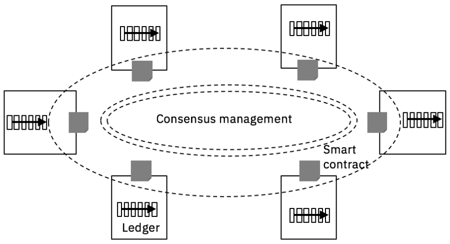

Fig. 2. A conceptual blockchain architecture.

Figure 3 presents the schematic design of our system as applied to a large-scale datacenter, comprising of multiple hardware devices each equipped with a blockchain chip on its motherboard. Each blockchain chip acts as a client to the system-wide blockchain service, which implements cross-system provenance and smart contract evaluation. These smart contracts can be derived from the technical and business specifications of individual hardware vendors, system integrators, application service providers, as well as end users of the computing system. The blockchain service both receives monitoring data from individual client chips, and sends instructions to the clients about what policies to use for data gathering. For example, data aggregated at the blockchain service may detect a high bit error rate in a specific model of memory chips, and this may trigger updating the policy in blockchain chips to monitor the operation of those memory chips more closely.

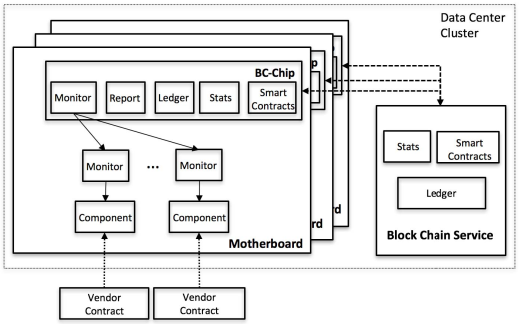

Fig. 3. Example System schematic for a data-center.

Figure 1 illustrates the architecture of the blockchain chip in more detail, and Figure 4 shows the functional process flow in a blockchain chip. The main functionality of the blockchain chip is to monitor and report operational information of some or all individual components on the motherboard. The chip collects data with a specific data collection policy that can be dynamically configured by the blockchain network and smart contracts. It contains a local private ledger where it stores the internal events and data, and which maintains trusted provenance for measurements collected. The chip based on the privacy policy, anonymizes the data collected and sends it out to the blockchain network. It further has a statistics module that can be used for local data aggregation, and it can be configured with policies that govern the frequency and granularity of data collection and reporting. The data collection policy specifies the rate, time interval, the amount, the type, granularity, and source of data being collected.

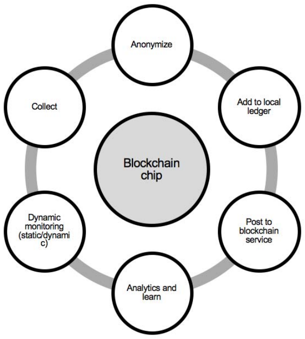

Fig. 4. Blockchain Chip Functional Process.

Motherboard-level blockchain: The blockchain chip also implements a blockchain service at the motherboard level, as illustrated in Figure 5, with the peers representing the technical/business administrators or stakeholders belonging to the vendor responsible for each chip, component, and/or shared resource on the motherboard that the blockchain chip is monitoring. Additionally, each peer has a smart contract deployed on the blockchain. There are smart contracts created as per the specification of each chip, component, or shared resource. The smart contract also represents the business understanding, the QoS levels, and the level of reliability, as well as the resiliency of the component they represent. The smart-contracts help determine the root cause of any specific problem that occurs on the chip, or whether the motherboard is working as desired, or if there are components that do not adhere to the desired QoS levels. If the root cause is not perfectly determined, an estimation of the cause may be provided. The smart-contracts can also define analytics and learning capabilities. The blockchain service engages in evaluation of smart-contracts and in consensus with respect to the problem associated with the hardware chips or components. The service makes a decision regarding what componentry and factors are responsible for any outage or issue with the motherboard. Consensus protocol ensures that the vendors and devices agree on certain events or output of analytics.

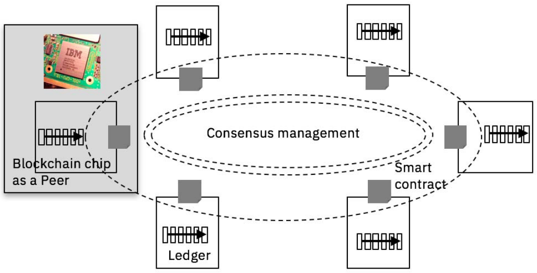

Fig. 5. Blockchain Chip Peer.

System-wide blockchain: The information from the motherboard-level blockchain service is propagated to another blockchain network (system-wide) managed at the higher level of the system hierarchy (e.g. datacenter rack, aisle or whole datacenter level). In the system-wide network, peers outside the motherboard can join and facilitate future audits by including data in its ledger. This scenario of the system-wide blockchain network is illustrated in Figure 6. At the system-wide level, cumulative violations of contracts across devices or chips can be evaluated. Further an aggregated decision can be made on how and which chips malfunction or whether there is undesired behavior of some chips, and corresponding actions can be taken. If a chip is not behaving according to the smart contract on the blockchain service, an alert is raised. A smart contract sends a request to increase or reduce the frequency of reporting by the blockchain chip based on whether there are smart contract alerts and statistics. Also, the precise nature of the information reported or added to the block may change based on an estimated risk (e.g. a risk of failure, or a risk of security breach).

The system is adaptive and can dynamically determine when to switch on and off the recording of certain parameters by the blockchain chip on a given hardware component by learning about the events and states in other hardware components.

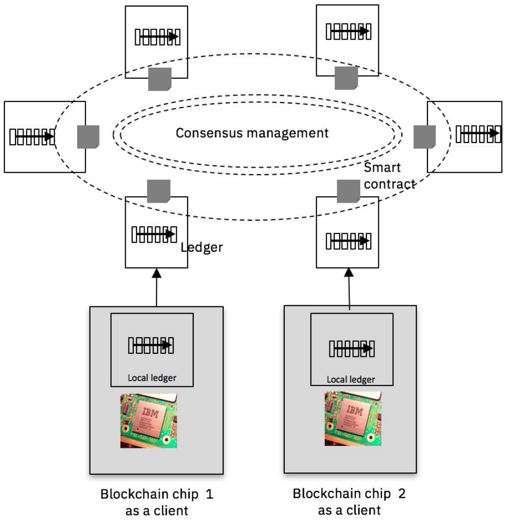

Fig. 6. Blockchain Chip Client.

In addition, the blockchain-based system includes privacy-protection and security functionality. It (1) ensures that only anonymized data goes out to a blockchain network that cannot receive private data; (2) ensures that sensitive data is sent to only the (private) blockchain ledgers and network that are authorized to receive such data with parties regulated and trained appropriately; (3) ensures that data is collected only when needed and configured to do so (need-to-know policy is implemented); and (4) helps detect and mitigate security issues in the hardware components. For determining whether there is a trojan or malicious backdoor in the system, the hardware device circuitry can use PUFs (Physically uncloneable functions) [9] to determine if there is any trojan chip/circuitry built into the system. Certified chips, i.e. chips with digital identities signed by vendors and their certificates, can also be used to ensure that trojans or unauthorized chips are not included in the system. Integrity of the circuitry and system can be validated against the signatures and/or PUFs [10]. Blockchain receives the PUF related data and security specific parameters and analyzes them to develop metrics such as Indicators-of-Compromise [11].

The information captured by the blockchain service can also be used to estimate the utilization and criticality of different features of a component as it operates within a complete deployed system. Industry peers, acting individually or collectively, can use this information to improve on cost/efficiency of their implementations and to evolve interface protocols taking into account actual requirements in deployment.

In summary, our proposed platform supports some of the vision goals as follows:

- Our system allows chip vendors a way to encode the technical specs of their products and a way for their clients, for instance datacenters, to encode their business contract with vendors.

- Our system supports immutable provenance that can be trusted by all parties/vendors.
- Our system offers a solution at the scale of a datacenter or city-wide IoT devices by leveraging a blockchain service/network that co-ordinates across the solution-components in these application scenarios.
- The consensus protocol deployed as part of blockchain can be used to ensure that various parties agree with the data, events or actions based on smart contracts.
- Our system offers a platform to conduct forensic analysis of data from legers by leveraging the audit capability of blockchain.
- Multiple blockchain networks can be deployed and a given hardware device can be part of some or all such networks at the same time. One network may focus on operating states whereas another network may focus on faults and another network may focus on security issues.
- The block chain chip interfaces with the monitoring system on the motherboard, with smart contracts and with the blockchain service.

V. Related Work

Blockchain of blockchains and hierarchical blockchain concepts have been explored by others in the scientific community and industry [12], [13]. Automation of policies using smart contracts has been proposed for bitcoin [14] and blockchain [15], [16], [17].

Mellis et al. [18] discuss how open-source hardware development worked for Arduino Duemilanove [19]. [20] presents the security and privacy challenges in IoT devices in the enterprise. [21] discusses a hardware chip enabling blockchain for Bitcoin and blockchain transactions to be signed and verified (Filament chip).

In software systems, and machine learning systems using machine learning artefacts, one can extend the vision and apply our proposed solution in order to manage the lifecycle, monitoring and security of the system [22]. A blockchain component replacing the blockchain chip carries out the tasks of the chip. Similarly, energy utility management can be carried out by using blockchain [23].

The Open Hardware Monitor [24] is a free open source software that monitors temperature sensors, fan speeds, voltages, load and clock speeds of a computer. The Open Hardware Monitor supports most hardware monitoring chips found on mainboards today. It supports monitoring Intel and AMD processors, the sensors of ATI and Nvidia video cards as well as SMART hard drive but it requires an OS to work. Computer-component power-consumption monitoring and control, described in [25], is a system that monitors and regulates the power consumption of various add-on cards and devices in a computer system by dynamically adjusting the device parameters such as clock speed, wait states, etc. A monitoring system for memory integrity is presented in [26]. A host system integrity monitor for monitoring memory, operating systems, applications, domain manager, etc. is isolated independent of the CPU and operating system. This monitoring system can be an add-in card or a coprocessor and it computes the hash values of the monitored systems, compares it with expected hash values and sends reports of discrepencies.

Security researchers have found that viruses can burrow in to a computer's motherboard, infect PCs as soon as they boot up, and are particularly difficult to detect and dispose of, for instance Trojan.Mebromi [1], a piece of rootkit malware malicious software that hides its presence on infected systems. It worms its way into the basic input-output system (BIOS) built into a computer's motherboard. Modified computer motherboard security and identification system [27] is a system to ensure no untrusted device is attached to the computer system and keeps monitoring the state of the trusted OS. This invention can also be used to allow identification and authentication of the computer and its users in networks. Chip integrity has been discussed in [10].

VI. Conclusions and Future Work

Hardware devices and systems have been treated as black-boxes in most scenarios such as in the context of design, testing, verification, monitoring, problem determination, problem prediction, and trust assurance. Governance of such processes for a hardware system or a collection of hardware systems requires understanding the internal states of the hardware, determining whether they adhere to the respective technical and business specifications and policies, and analyzing and learning from them.

In this paper, we discussed how hardware governance is going to change in future, and what are the features needed to enable a transparent, auditable, scalable, secure ecosystem for white-box hardware governance. The systems platform in our vision should be developed such that it (1) provides a means to formalize and enforce requirements for individual components; (2) provides both component-vendors and consumers (e.g. datacenter operators) a trustworthy mechanism of establishing claims against contracts; (3) allows timely and deterministic resolution of cross-component system issues; (4) can quickly determine source(s) of problems with provenance tracking; (5) can ascribe responsibility with automatic consensus; (6) can help in automatic re-imbursement/replacement of components as per contract; (7) can facilitate system-wide monitoring, replacement of hardware components and claims; and (7) mitigates risks for system integrators building complex systems.

We presented the research challenges and issues associated with realizing this vision. We also presented our approach to building an ecosystem using the notion of a blockchain chip and blockchain platform. In the future, we plan to extend this work to address some of the research challenges mentioned in this paper.

Acknowledgement

We would like to thank Cliff Pickover for his inputs and feedback on the contents of the paper.

References

[1] "Trojan.mebroni," http://www.nbcnews.com/id/44554808/ns/technology_and_science-security/t/nasty-new-virus-infects-your-pc-motherboard/#.WZ33M2dGGxg".

[2] "Openpower foundation." [Online]. Available: https://openpowerfoundation.org

[3] "Open source hardware association," 2018. [Online]. Available: https://www.oshwa.org/

[4] "Open connectivity foundation." [Online]. Available: https://openconnectivity.org

[5] A. Waterman, Y. Lee, D. A. Patterson, K. Asanovic, V. I. U. level Isa, A. Waterman, Y. Lee, and D. Patterson, "The risc-v instruction set manual," 2014.

[6] A. Moradi, A. Barenghi, T. Kasper, and C. Paar, "On the vulnerability of fpga bitstream encryption against power analysis attacks: extracting keys from xilinx virtex-ii fpgas," in *Proceedings of the 18th ACM conference on Computer and communications security*. ACM, 2011, pp. 111–124.

[7] G. P. Sutton and O. Biblarz, *Rocket propulsion elements*. John Wiley & Sons, 2016.

[8] M. Swan, *Blockchain: Blueprint for a New Economy*, 1st ed. O'Reilly Media, Inc., 2015.

[9] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, 2002.

[10] M. Lecomte, J. J. A. Fournier, and P. Maurine, "On-chip fingerprinting of ic topology for integrity verification," in *Proceedings of the 2016 Conference on Design, Automation & Test in Europe*, ser. DATE '16. San Jose, CA, USA: EDA Consortium, 2016, pp. 133–138. [Online]. Available: http://dl.acm.org/citation.cfm?id=2971808.2971838

[11] J. T. Luttgens, M. Pepe, and K. Mandia, *Incident response & computer forensics*. McGraw-Hill Education Group, 2014.

[12] "Polkadot's plan for governing a blockchain of blockchains," 2018. [Online]. Available: https://www.coindesk.com/polkadots-radical-plan-governing-blockchain-blockchains/

[13] "Internet of blockchains," 2018. [Online]. Available: https://cosmos.network/

[14] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[15] C. D. Clack, V. A. Bakshi, and L. Braine, "Smart contract templates: essential requirements and design options," *CoRR*, vol. abs/1612.04496, 2016. [Online]. Available: http://arxiv.org/abs/1612.04496

[16] J. Sun, J. Yan, and K. Z. Zhang, "Blockchain-based sharing services: What blockchain technology can contribute to smart cities," *Financial Innovation*, vol. 2, no. 1, p. 26, 2016.

[17] D. Tapscott and A. Tapscott, *Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world*. Penguin, 2016.

[18] D. Mellis and L. Buechley, "Collaboration in open-source hardware: third-party variations on the arduino duemilanove," in *Proceedings of the ACM 2012 conference on computer supported cooperative work*. ACM, 2012, pp. 1175–1178.

[19] "Arduino duemilanove," 2018. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoardDuemilanove

[20] A. Sadeghi, C. Wachsmann, and M. Waidner, "Security and privacy challenges in industrial internet of things," in *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2015, pp. 1–6.

[21] N. De, "Iot startup filament hits milestone for blockchain hardware." [Online]. Available: https://www.coindesk.com/iot-startup-filament-hits-milestone-for-blockchain-hardware/

[22] J. M. Castro, "Software asset management use case for blockchain," 2017. [Online]. Available: https://www.ibm.com/blogs/insights-on-business/electronics/blockchain-for-software-asset-management/

[23] P. Maloney, "Blockchain could change everything for energy," 2018. [Online]. Available: https://www.renewableenergyworld.com/articles/2018/02/blockchain-could-change-everything-for-energy.html

[24] "Open hardware monitor," http://openhardwaremonitor.org/, accessed: 2016-11-30.

[25] D. F. Hepner and A. D. Walls, "Computer-component power-consumption monitoring and control," Nov. 2006. [Online]. Available: https://patents.google.com/patent/US7134029B2

[26] W. A. Arbaugh, N. L. Petroni, T. J. Fraser, and J. M. Molina-Terriza, "Method and system for monitoring system memory integrity," Feb. 02 2015. [Online]. Available: https://patents.google.com/patent/US20090217377A1

[27] J. A. Tello, "Modified computer motherboard security and identification system," Oct. 10 2002. [Online]. Available: https://patents.google.com/patent/US6463537B1