# BLOCKCHAIN ANALYSIS NODES DEFENDING AGAINST INTRUSION AND DENIAL OF SERVICE

INVENTOR:

JASON W. RUPE

# Blockchain Analysis Nodes Defending Against Intrusion & Denial of Service (BANDAIDS)

## Description

Blockchain technologies are being used for many things, and that usage is growing exponentially. The applications are often critical, or at least significant in value. The stability of these systems are questionable, and there is significant evidence that these systems are not guaranteed stable. Further, we know they are susceptible to attack, even though they are often designed to guard against these attacks. Because they are dynamic systems, and incredibly complex with many uncontrolled factors, there can be no inherent guarantee. So, some external system of support is necessary for some applications.

If we put safeguards in place, we could improve the security, safety, and reliability of blockchain technologies.

At creation, these systems are often modeled and analyzed for risk. We can leverage the risk profiles and math describing the expected behavior to set thresholds for reporting and deeper scrutiny, create an overlay mechanism that further validates and verifies the health of the chain by searching for anomalies (statistical, logical, or otherwise) or breaks and bends in rules. This construct could form a framework for future AI and anomaly detection methods in this space. The system could include financial and risk measures for parties in consideration of participation, and provide estimates of risk coverage as an assurance. The system would include random re-verification, scans for bad players, scan for statistical anomalies and cheating (including subgroup collusions), scan for intrusions, and DoS.

Blockchain technologies are being used for many things, and that usage is growing exponentially. The applications are often critical, or at least significant in value. The stability of these systems are questionable, and there is significant evidence that these systems are not guaranteed stable. Further, we know they are susceptible to attack, even though they are often designed to guard against these attacks. Because they are dynamic systems, and incredibly complex with many uncontrolled factors, there can be no inherent guarantee. So, some external system of support is necessary for some applications.

Blockchain technologies are complex system with dynamic behavior. The study of these systems is in infancy, but developing. By applying known systems engineering knowledge, operations research, and the attack vectors and failure modes already identified, we could build a system that monitors and adds assurance to these systems so that they can be used in more critical application areas (IoT, medical, supply chain).

The key assertion that must be tested is whether such a supporting system could be defined, designed, built, and implemented to be effective. There are several unknowns to discover in this process.

- While some of these systems exhibit conditions that could indicate risk, it is not known in all cases if that is true, so systems analysis is in order.
- For systems that have been analyzed to some extent (and at times published in papers), can an overlay system be built to monitor for the identified indicators of these risks without impacting

the system's function, and without significantly changing the cost model of the overall solution (is the cure worse than the disease)?

- Can we define a general framework that works on a broad range of distributed technologies?

By creating this added layer of assurance in blockchain technology implementations, we improve the integrity of these systems, and thus the safety, security, and reliability of the systems depending on these technologies.

A decoupled solution such as proposed here is attractive because it could be created generally, applied as warranted to specific cases, and could be used in many different blockchain technology solutions.

Considering the space of this invention to be blockchain and related technologies, there are a huge number of systems that fit this area, and they mostly try to assess their systems for threat vectors, then prove the risk is low. In many cases, there are ad-hoc or to-be-determined methods for mitigation of system failures, because ultimate system failure will mean collapse of all dependencies, including trust.

Mostly, however, there are no outside mechanisms that completely close the loop on risk, or mechanisms that assess risk conditions and take measures to mitigate even when the risk was not predicted. The technology exists to create software systems that can assess and mitigate risk in these systems, but there is no evidence that such solutions being fully utilized.

As we have seen in these systems, when problems are observed, there is usually a mad scramble to recover, at significant cost, and human intervention to change the system in hopes of reducing overall risk, though not always with assurance.

All these systems carry risk, and not all risks can be inherently managed as there are always outside effects that can’t be controlled. Of these risks, there are some that can be identified and estimated under certain assumptions, and those that can’t. As such, there is room for improvement by reducing the known risks, and placing methods that search for unidentified risk.

If a risk vector can be identified, and the impact described, then the system can be monitored in some way so that a severity of risk can be measured. That measurement can then be assessed for relative severity, and actions triggered on it. Even in cases where the absolute severity of risk is not understood, such that the measure can only be identified as an anomaly, then the anomaly can trigger a search for other anomalies and evidence for cause analysis.

This invention proposes a companion system (cs, BANDAID) for all blockchain and related technology systems (bct). The companion system would do several things:

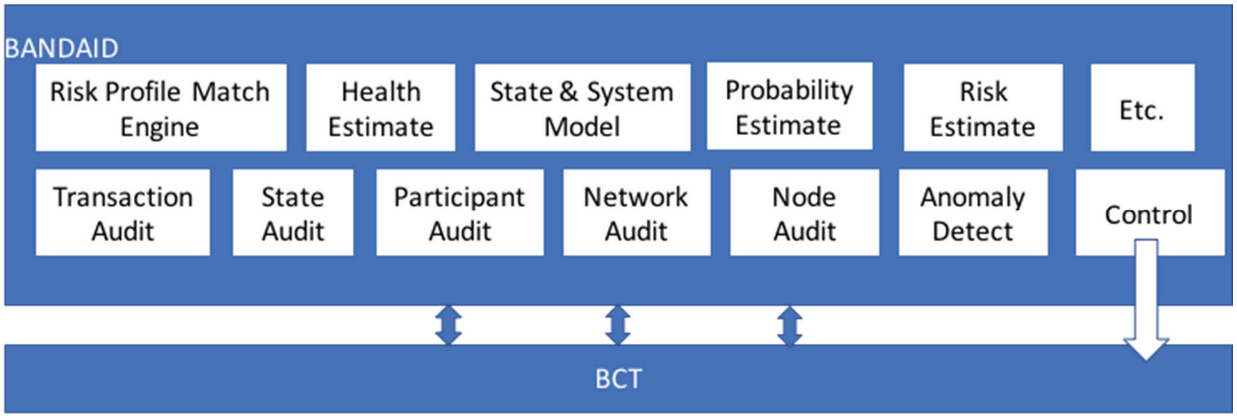
1) Remain independent, decoupled from the bct.
2) Monitor the bct for all known risks.
3) Monitor the bct for anomalies.
4) Model the bct to match risk conditions and provide predictions of future possible states or possible causes, plus assessments of probability that the bct is healthy.
5) Audit the bct for overall health, latent attacks, etc.
6) Exert external controls to mitigate risk when possible and reasonable.

While many bct today have assessed their identified risk vectors, they don't have independent, decoupled systems that measure the risk state of the bct, search for problems, search for anomalies, trigger follow up actions to search for causes of the risk state, or put emergency measures in place to force system correction.

The possible controls could include but are not limited to the following:

- Increase the required redundancy of calculations
- Additional validation steps
- Conditional audits
- Throttling speed of the processing
- Adjust to other bct parameters
- Removal of participants
- Invalidating transactions and setting back bct state

**Diagrams of system**



Such a system should be implemented in software, though it may be implemented in part through procedures or processes.

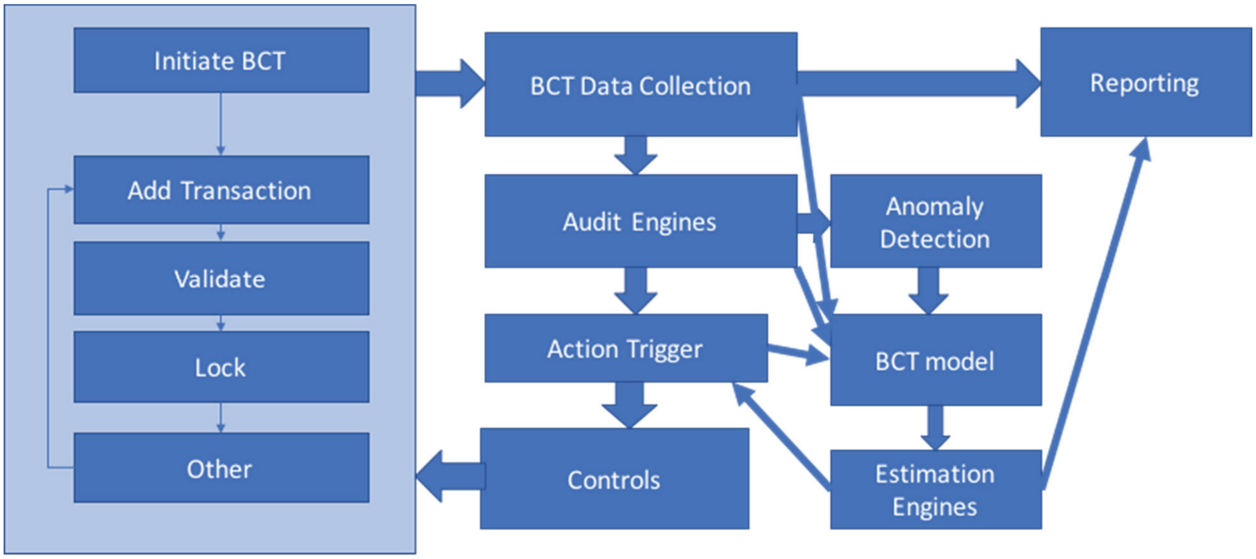
It could be implemented in a cloud architecture.

It could be built to be redundant.

It could be a collection of modules or functions or objects that work independently, in concert, hierarchically, or as a network, or any combination of these.

It may have a user interface for control, or to facilitate risk assurance of the bct.

**Flow Diagram**



**Example**

Consider a general bct that exhibits an indicator of a known risk vector. The BANDAID system correlates the indicating statistic with additional information to support the indicated risk vector, and estimates the strength of indicating that there is a problem requiring action. If an action is indicated, it is triggered. One possible finding may be that a transaction is not valid, in which case the invalid transaction is removed, and the bct reverts to an earlier state. Another possible finding is that there is a participant that is misbehaving, in which case they are removed and potentially corrected. Another possible finding is that the state is valid but anomalous, so additional checks may be in order, or at least the risk profile of the bct is reported.

**Example risk vectors include but are not limited to:**

- One party having significant weight on the network such that they are more able to exert control in unwanted ways.
- Any double spend on the network, or contracts in conflict.
- Any network statistic that represents unlikely system state, such as a set of transactions that appear more separate from the network than usual, or a node that is proving too much work.
- A transaction rate that appears statistically unlikely (too frequent, too infrequent, or of atypical size, etc.).
- Any atypical rate of joins or leaves on the network.
- Any measure that is highlighted in the risk analysis for a bct.
- Any cost or benefit that is abnormally high.

For example, for IOTA, this would include anyone with an abnormally high amount of computation power, or any evidence of collusion among participants, any participant behaving outside their best interests, any large weight attack, any weighting that is outside expectations, any evidence of a parasite chain or split forming, large numbers of lazy tips, or significant tip selection results that deviate from statistical expectation.

**Methods for identifying triggers include but are not limited to:**

- Typical statistical quality control methods such as control limits or specification limits based on risk models or baseline statistics including X-bar, R, p, EWMA, etc. Standard methods for calculating statistics and examining outliers applies here.
- Any condition that indicates a known problem including a double spend attempt. Typical search algorithms and sampling techniques apply here.
- Any of a number of methods that detect anomalies, or detect signals in noise, such as those used for Prognostics and Health Monitoring (PHM).

Each bct has features that are common which should be benchmarked and could be indicators of risk when they change rapidly or outside a baseline expectation. Some of these features are known to equate to risk, and others are not but may be important indicators of risk nonetheless.

Flows:

1. Identify the elements of the bci, and how to measure each quantifiable feature.
2. Establish a measurement method for each feature, and collect the data using a given management data architecture that applies.
3. Feed each data element into several different analysis engines including but not limited to
    a. Statistical quality and specification control: X-bar, R, S, P, NP, etc.; one sided and two sided included as reasonable.
    b. Analytics: display each feature, and automated correlation analysis.
    c. Prognostics: apply known models to find signal in the noise to indicate a change in behavior, and predict the “lifetime” of the system.
    d. Anomaly Detection: generalized data feature identification and correlation to start.

**Self-revision**

For features identified as risk factors which lead to desired changes in the bct, the BANDAIDS can influence the operation of the bct, if desired. Inherent in existing systems like Bitcoin are controls that change the amount of work to close a block; by specifying the number of 0s in the solution, the amount of work is controlled. Likewise, nodes could be removed from play for a duration based on lack of trust due to certain risk factors, or certain transactions may be required to be added to the next block, or limits on the items to validate may be placed into the system. This added concept relates to Cognitive Elastic Networks (D3418).