Trustworthiness Model Representation

Authors:

Vincent Bemmel
Industry Technology Lead
Corlina
vincent@corlina.com

Sameer Merchant
Chief Technology Officer
Corlina
sameer@corlina.com

Shashi Sastry
Chief Product Officer
Corlina
shashi@corlina.com

Amy Hawman
VP of Marketing
Corlina
amy@corlina.com

INTRODUCTION

Trust is a desired attribute of all connected systems in business. But with the advent of mission-critical Industrial Internet of Things (IIoT) systems that control the production of a company's final output to their customers, the importance of a trusted system has been magnified: Without a trustworthy system, a business cannot have confidence that it will meet its customers' needs, its legal and regulatory obligations, and, ultimately, its business objectives.

Trustworthiness is defined by the Industrial Internet Consortium (IIC)[1] to be a composite of attributes of safety, security, privacy, reliability and resilience. This is an essential definition for "what" trust is in the context of an IIoT system. These individual characteristics of a trusted system are interdependent and, at times, inversely related. As such, measuring trustworthiness can be very subjective based on the specific goals of an organization and the implicit trade-offs it has made. For an effective approach to 'how" an organization actually implements a policy to ensure trustworthiness, it is imperative that a model for measuring trustworthiness must be both rigorous as well as pragmatic, able to address the specific operating concerns of each organization. For example, a systemic measure of trustworthiness may rely more heavily on security attributes in a large, distributed industrial environment such as energy production, than in an access-controlled production facility such as food manufacturing, where (product) safety may be more important.

Core aspects of measuring and reporting trustworthiness include 1) providing model users with near real-time visibility into the IIoT system status, 2) providing assurance that the system is operating according to specifications as well as providing immediate information on changes to the system, 3) providing certification of the trustworthiness and authenticity of new system components as they are added to a growing infrastructure and 4) providing a record of how the system status has changed over time.

Creating a robust yet practical Trustworthiness Model must therefore account for the context of the operating environment of the system itself and the unique priorities of the organization. And it must be intuitive and actionable so that organizations will come to rely on its usefulness, rather than seeing a Trustworthiness Model as an interesting yet academic measure with limited value to daily operations.

BACKGROUND

Trustworthiness Defined

An IIoT system needs to be trustworthy in order to gain the trust of its users. According to the Oxford English Dictionary, trust is defined as a "firm belief in the reliability, truth, ability, or strength of someone or something." While trust is a logical evaluation of someone or something, it is

[1] The Industrial Internet Vocabulary Report, page 21, https://www.iiconsortium.org/pdf/IIC_Vocab_Technical_Report_2.1.pdf, 2018.

also an emotional evaluation – and understanding this is essential for modeling the trustworthiness of a system. An effective model must therefore address the emotion as well as the logic, through transparency, visual reinforcement of results and consistency over time -- all supporting the visceral response of the user in addition to the inherent logic of the model itself.

Another interesting property of trust is that it is contextual – out of a complex set of variables, people have the ability to compartmentalize and invest their trust in what matters most.

In the context of a cyber-physical IIoT system[2], the IIC defines trustworthiness as "the degree of confidence one has that the system performs as expected with characteristics including safety, security, privacy, reliability and resilience in the face of environmental disturbances, human errors, system faults and attacks." But at its most basic level, trustworthiness reflects the convergence of a set of fundamental *system* characteristics that collectively measure these five properties, and these system characteristics were traditionally separated between two user groups (Information Technology (IT) and Operational Technology (OT)). This means two different sets of audiences with different perspectives, potentially looking at the same set of metrics.

The challenge is to focus on what matters most in order to evaluate trust in the context of a given IIoT solution and audience. There is no single way to represent trustworthiness, and a flexible scheme is required to adapt to the relevant context. Without that, one can end up with an overly complex and noisy model that does not intuitively reflect the key aspects of trust as they matter to the observer of the system.

A user will be able to maintain or enhance their trust in a system, when she has confidence in the following, among others:

- the system performs as designed, and continues to do so throughout the lifecycle of the system (through continuous verification/visibility)
- the system has historically performed as expected, and disruptions were minimal (through historic evidence and documentation)
- the system and the data it is producing are authentic and has not been tampered with (through continuous trust certification)
- the system/vendor has a good reputation

Most of these items can serve as a blueprint for building a trustworthy system.

The trustworthiness of IIoT systems can be evaluated by decomposing them into smaller components and then evaluating the individual trustworthiness of each component as it contributes to that of the overall system, in a way similar to that

[2] NIST Special Publication 1500-201, https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1500-201.pdf, June 2017.

described by NIST[3],[4]. This approach greatly simplifies the implementation of trust verification systems.

Practical Considerations

Establishing Trustworthiness requires a Trust Model and a Trust management system or framework. Trust as a computational concept has been discussed for many decades. Stephen Marsh formalized Trust as a computational concept in 1994[5]. Since that time, many Trust models and frameworks have been proposed.

Of particular note is the framework introduced by Carmen Fernandez-Gago, et al, for developers to incorporate trust in IoT solutions[6]. The proposed framework addressed trust, privacy and identity requirements for inclusion of trust in the IoT. Dario Ruiz Lopez, et al, also proposed a trust model and a framework. In their paper, they highlight the importance of providing a clear method to interpret and act on alerts from a trust system[7]. And David Maher has called for a human-centric trust model for the Internet of Things. In his article, Maher suggests "For IoT security to be successful, there needs to be an effective way to reason about how humanity can trust the security, safety, and privacy of this massive transformation of the world."[8]

In this paper, we attempt to provide a human-centric approach to establishing trust for IIoT systems. Our proposed model and framework endeavors to provide an average user with a reasonable understanding of the integrity of their connected devices.

Keep in mind that the main users of IIoT systems (e.g., in a connected smart factory) are in many cases very pragmatic, mostly driven by keeping the system running in order to meet production goals. They are often just looking to add simple confirmation of trust to their legacy situational awareness tools (addressing a common blind spot today). It is therefore important that trustworthiness models result in simple informative tools with minimal additional complexity. A trustworthiness representation has to be practical and intuitive in order for it to be valuable.

Here are a few key guidelines to be considered for an effective representation:

- Context matters – depending on the application, different aspects contribute to trustworthiness. An effective presentation will allow for customization to include/exclude aspects that are important to the user.

[3] Brian A. Weiss, Michael Sharp, and Alexander Klinger, "Developing a hierarchical decomposition methodology to increase manufacturing process and equipment health awareness," https://ws680.nist.gov/publication/get_pdf.cfm?pub_id=925101, 2018.
[4] E. R. Griffor, "Toward a Calculus for Optimizing CPS to Trustworthiness," http://www.ices.kth.se/upload/events/172/a77727a10d7f4a07b982404fab4effc8.pdf, 2017.
[5] Stephen Paul Marsh, "Formalising Trust as a Computational Concept," https://www.nr.no/~abie/Papers/TR133.pdf, 1994.
[6] Carmen Fernandez-Gago, Francisco Moyano, and Javier Lopez, "Modelling Trust Dynamics in the Internet of Things," https://www.nics.uma.es/pub/papers/Fer_IS17.pdf, 2017.
[7] Dario Ruiz Lopez, et al, "Modelling the trustworthiness of the IoT," https://www.researchgate.net/publication/308928730_Modelling_the_trustworthiness_of_the_IoT , 2016.
[8] David Maher, "A human-centric trust model for the Internet of Things," https://www.oreilly.com/learning/a-human-centric-trust-model-for-the-internet-of-things, 2017.

- Trust is subjective – it addresses a deep emotional aspect of humans. Hence, trustworthiness solutions should be capable of approaching the observer from that angle.
- At the same time, humans have been shown to exhibit bias in the creation in their subjective assessments of trust, as shown by Fawcett, et al, in their study on the nature of trust in buyer/supplier relationships[9]; in order to be effective, the representation must eliminate the inherent biases by relying on explicit measures.
- Operational history of a system facilitates establishing the trust in a system, while continuous visibility can be re-assuring and help maintain that trust.
- It should be simple, intuitive and relevant to the user.

Although trustworthiness is composed of various characteristics, a simple, intuitive representation is essential. With the key context preserved, such a model can quickly be comprehended, making it very effective.

This is possible, as will be represented in the following section. A management tool can provide a trust indicator as proxy for representing the trustworthiness of a system. In addition to displaying key indicators, support for a hierarchical decomposition helps the observer quickly navigate to different aspects of a system where trustworthiness may be compromised.

Creating a Model to Reflect the Definition

Once the key elements of trustworthiness are identified, the next step is to develop a representation model.

An adequate trustworthiness model has to:

- Be flexible to include or filter out metrics, based on the target context and audience
- Account for the history of the system, and allow trust to improve with time (e.g., impacts of an incident shall decay over time to reflect restoration of trust)
- Support hierarchical decomposition in order to evaluate impacts of trustworthiness in different layers of the system
- Facilitate a simple representation/visualization with relevant information

A common method to visualize trustworthiness is via a 'radar' or 'spider' chart, where the five definitional characteristics are represented via individual scores on each axis. Each of the characteristics is composed of a varying set of capabilities, and their contribution is then scored/normalized and summed over the associated axis. This method of visualization is intuitive on a high level.

[9] Stanley E. Fawcett, et al, "I know it when I see it; the nature of trust, trustworthiness signals, and strategic trust construction," https://www.emeraldinsight.com/doi/full/10.1108/IJLM-11-2016-0268, 2017.

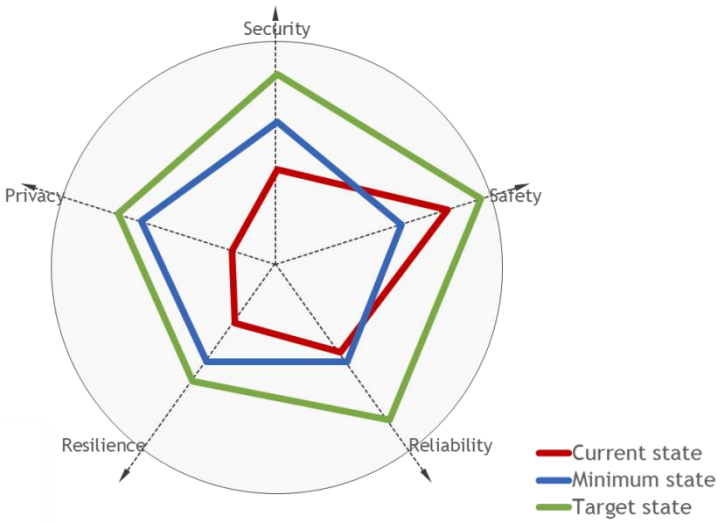

Figure 1: Trust Model Spider Chart

We propose a model that allows for inclusion of the relevant contributing factors, while enabling the user to selectively emphasize important factors and deprecate those of less importance. This model can then be used to represent the overall solution trustworthiness as a cumulative single score. The proposed model does not mandate using just the single cumulative score: Users can still visualize individual Trust Scores using a 'radar' or 'spider' diagram. We will demonstrate such a model in the following section.

THE MODEL

As noted earlier, trust may mean different things in different applications. Trust in a home security system may require all components involved in monitoring the perimeter of a house to be trustworthy. Home security system trust would involve ensuring all security system hardware and software components are trustworthy. Additionally, the solution requires that the security system configuration, the location of all the sensors involved and the observations made by the system can be trusted.

Trust in the food industry may involve monitoring a different set of parameters. It may require trusting the production, shipping and storage process. For example, a consumer can trust a food product provided the source of the ingredients, the production process, the packaging, transportation and storage can be trusted and verified. Thus, each application may require a different set of parameters to establish trust.

Trustworthiness of a solution requires measuring the Trust Score of each component or entity involved in delivering that solution. The overall trustworthiness of a solution can then be a weighted combination of the Trust Score of each element of the solution.

Components of Trustworthiness

Trust provides a measure of confidence. As discussed above, each application may involve a different set of parameters or

attributes that contribute to an overall trust measure. Hence it is essential to define a generic model to compute trust. The trustworthiness of a solution involves

- Computing a Trust Score for each component
- Computing an overall Trust Score that is a combination of the Trust Scores of each component involved in providing a solution.

The model used for computing a trust measure must be generic enough in order to provide a Trust Score for various applications. The IoT Policy Framework defined by EU commission calls for a similar hierarchical approach to establishing trust.[10]

The attributes involved in computing the Trust Score of a component can be classified into two broad classes:

A. **Quantitative attributes**: Such parameters can be measured numerically. Examples include:
   a. Temperature values measured by a sensor.
   b. Date, time and size of a file installed on the system.
   c. A fingerprint – like a secured hash of a file installed on a system.

B. **Qualitative attributes**: Such characteristics are assigned a unique label from a set of valid labels (i.e., classification). Typically, such attributes are assigned a string value like a label assigned to an object in an image observed by a camera. Examples could include:
   a. Vehicle type identification such as truck, forklift or unknown vehicle. A factory could use this kind of qualitative parameter to establish trust for delivery trucks. It would ensure that only authorized vehicle types are docked at the loading stations.
   b. Part type at an assembly work station. A factory could use this type of parameter to ensure that the correct components are used in an assembly operation. It ensures the trustworthiness and quality of the final product.
   c. Ingredient name/label at a mixing station. The food industry could use such an attribute to establish trust in the final product, providing for ingredient safety and product quality.

The attributes are further grouped into logical sets. As an example, in a manufacturing solution, the attributes of a component within a manufacturing solution can be grouped into the following logical sets:

- System hardware attributes
- System software attributes
- Application software attributes
- System access attributes
- System temperature attributes
- System vibration attributes
- System configuration attributes
- Observed pattern attributes

Each attribute can be assigned a different weight based on how it affects the overall

[10] H2020 – CREATE-IoT Project, https://european-iot-pilots.eu/wp-content/uploads/2017/10/D05_01_WP05_H2020_CREATE-IoT_Final.pdf, 2017.

Trust Score of a component. An application can assign different weights to attributes contributing to the trust measure. Further, an application can assign weights to logical sets of attributes.

The qualitative and quantitative parameters can be further mapped to the five broader categories of safety, security, privacy, reliability and resilience of the solution. Each parameter contributes towards computation of the trustworthiness of each of these attributes.

Representing Trustworthiness

An effective representation of trustworthiness is through a numerical score that provides actionable information to the user. Based on the Trust Score, the user can get answers to questions like: (a) Can I trust an entity? (b) Can I trust a solution? (c) What is the confidence level of the performance of an entity or a solution? (d) What factors were considered in measuring trust? (e) What affects the trust of an entity or a solution? (f) What actions should a user take to restore trust? This quantitative measure should help the user of a trust model focus on problem areas by knowing what impacts trust and enables the user to take corrective actions to restore trust. At the same time, this score will not account for elements such as policies and procedures that also contribute to the overall trustworthiness of an environment. Thus it is a component of an overall suite of tools that an organization will employ to ensure end-to-end trustworthiness, beyond that measured for its cyber-physical systems.

A numerical Trust Score is calculated by using an algorithm that combines individual scores of all observed attributes.

Trust Score Function

A Trust Score Function is responsible for computing the Trust Score of a solution based on a set of observed qualitative and quantitative attributes.

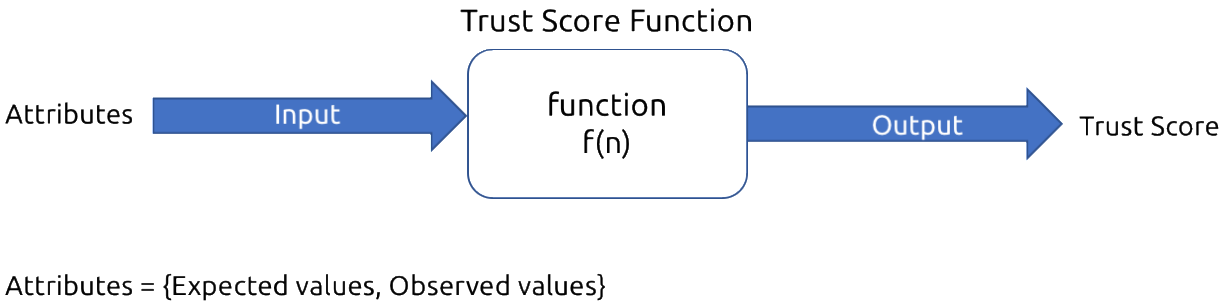

Figure 2: A Trust Score Function takes attributes as input (expected and observed values) and produces a Trust Score

The Trust Score Function defines the range of input values and their interpretation, and uses that to calculate the associated Trust Score. It may be preferred to compute a Trust Score on a linear scale, as a linear representation may be easier for users to interpret. A higher score would represent a highly trusted solution. A lower score would represent a lower level of trust in a solution.

The trust measure must also be explainable, i.e., the user should be able to determine what contributed to a particular Trust Score. This makes the Trust Score actionable. If required, a user can take corrective action by addressing the cause of a particular Trust Score.

dependency relation of components and characteristics that define a solution can be represented by a directed acyclic graph. A node in the graph represents a component or an observed attribute. A directed edge identifies the dependency of the relationship. An edge from a source node to

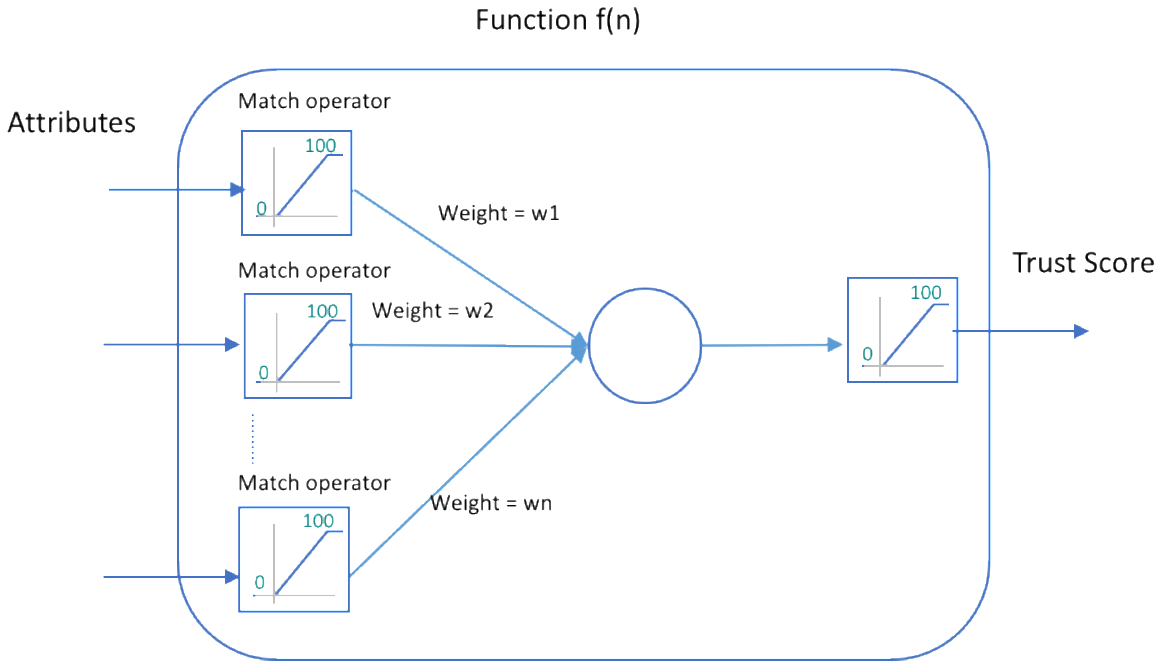

Figure 3: The relationship of attribute Trust Scores to calculating an overall system Trust Score

The above figure illustrates an example of a function that maps observed attributes to a linear Trust Score in a range of 0 – 100%. A Trust Score of 100% indicates that the system is highly trustworthy, while a score of 0% indicates that the system cannot be trusted.

System Trustworthiness

A System or solution typically involves multiple components. The overall trustworthiness of the system or solution depends on the trustworthiness of each of its individual constituent components. The

a destination node indicates that the Trust Score of the source node influences the Trust Score of the destination node. The Trust Score of a node (component or system) is computed as a weighted combination of the components or attributes that affect its score. A user or a system designer can determine how much weight each component or attribute has on a Trust Score of a system. The weight of the Trust Score of a node on its dependent node is defined as the weight on the edge between the nodes.

The following figure illustrates an example of a system and its components represented by

a directed tree. The root node represents the system, and the intermediate nodes in the tree represent the constituent components of the system. In the example, the system is comprised of components 1 through m. Each component has a series of individual observed attributes (attributes 1 through j in the case of component 1) that are used to compute the Trust Score of the component. The observed attribute 1 is weighted by the factor w1, and so on, through attribute j which is weighted by the factor wj to compute the Trust Score of component 1. The computed component Trust Scores are then fed into the overall function to compute the Trust Score of the overall system, using the component weightings w1 through wm.

Organizing components and attributes of a system or solution in a directed acyclic graph enables the user to control and filter how each component or attribute influences trustworthiness. Computing a Trust Score of each component helps identify factors affecting the overall Trust Score. The directed acyclic graph allows components to be grouped based on the high level categories such as safety, security, privacy, reliability, and resilience.

Benefits of the Trust Model

The benefits of a Trust Model constructed in this manner are that it flexibly addresses the goals of a variety of solutions. The Trust Model supports basic operations such as *fusion* and *discounting* as specified in the Wikipedia definition of a trust metric.[11] Its output is simple -- the component Trust Scores and the overall Trust Score provide a measure that is straightforward to interpret and enables the user to readily take action to address the root cause of issues that diminish trust.

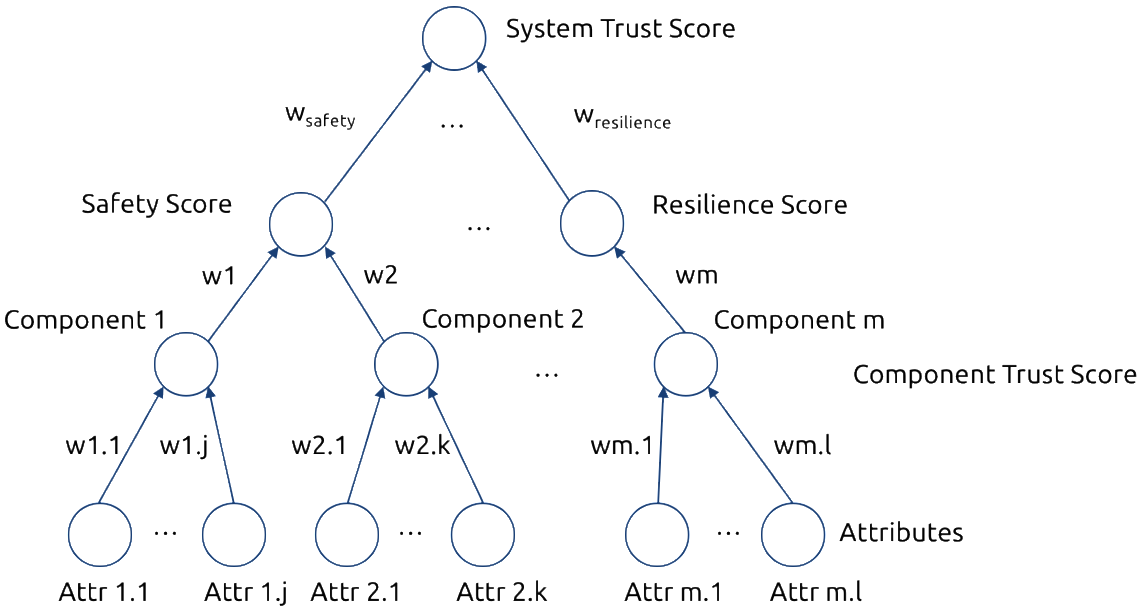

Figure 4: Computation of a Trust Score

[11] Wikipedia, https://en.wikipedia.org/wiki/Trust_metric, 2018.

The Trust Score model is designed to provide visibility into the factors affecting overall system trust. Representing trust as a hierarchical combination of component trust helps the user to determine all factors affecting a system's Trust Score. When a system Trust Score is low, the user can drill down into a specific (child) component with a low score and identify the root cause. As an example, the overall system Trust Score could be a weighted combination of two high level attributes, system security and system safety. The user can observe each attribute Trust Score and identify if the overall system trust is low due to safety or security issues (or both). The user can further drill down the graph to identify each factor affecting the safety or security attributes. The following figure illustrates how the trust model can facilitate the identification of the cause of a low Trust Score.

Figure 5: Computation of a System Trust Score with Security and Safety Attributes

MODEL APPLICATION

The Trust Model provides a quantifiable method, which when associated with the business data, measures the effectiveness of the business models and therefore the business outcomes.

While the model itself can be used to evaluate the trustworthiness of various system characteristics, in its application, we are focused on measuring the trust in the delivery of data from the data source to its destination. Calculation of the trustworthiness of the data requires that we have to measure all points - systems or devices generating the data, the network transporting the data, the communication links, the process of collecting the data, and the people associated in any or all of these points. The focus of this paper is to apply the

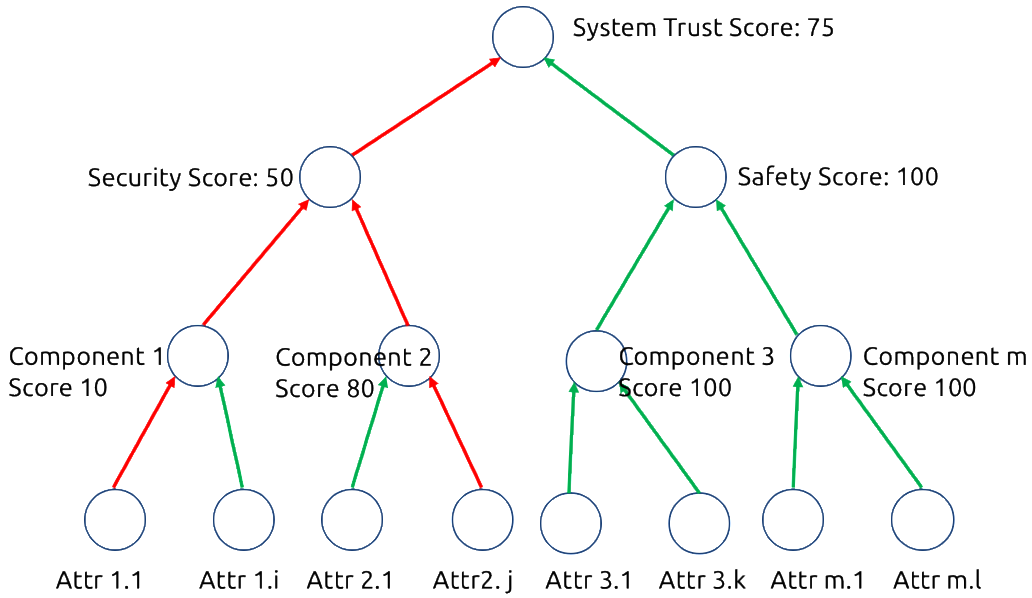

Figure 5: Computation of a System Trust Score with Security and Safety Attributes

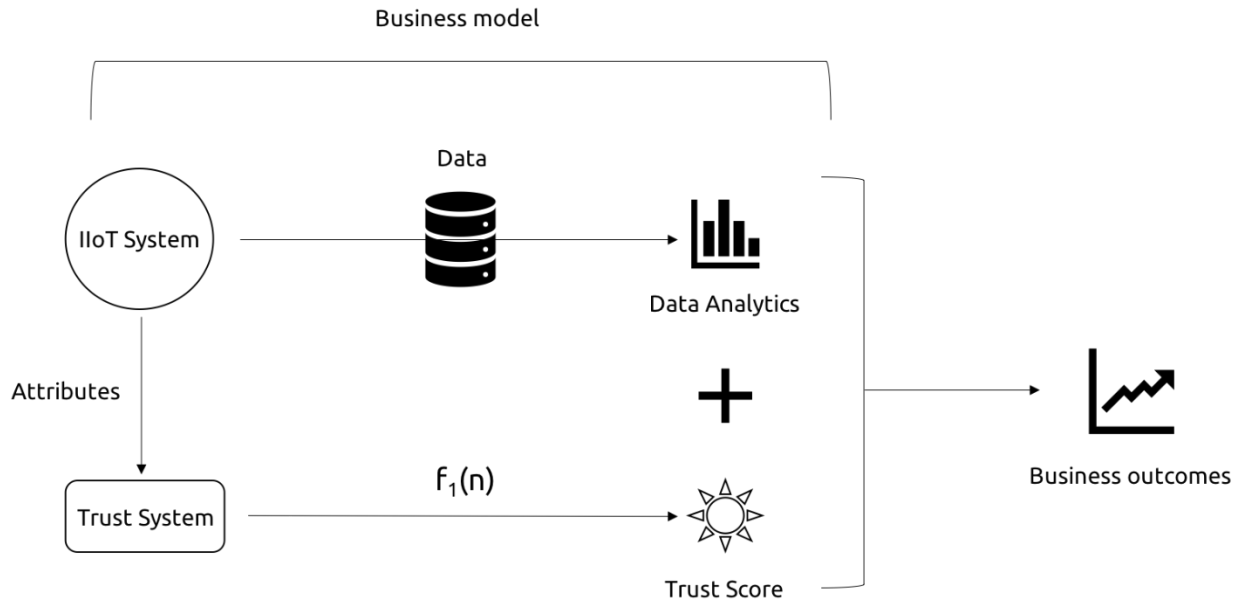

Figure 6: The Trust System quantifies the business model and increases confidence in the business outcomes

model to the device that is generating the data and the system that supports its generation and transportation to its final destination for analysis.

Trust is important to ensure data quality for data analytics. Any data quality management effort should start with collecting data in a trusted environment. This in turn implies that the data sources (machines, IoT devices, etc.) and the data collection processes are all trusted. Too often data analysts find that they are working with data that is incomplete or unreliable. They have to use additional techniques to fill in the missing information with predictions. While techniques such as machine learning or data simulations are being promoted as an elixir to bad data, they do not fix the original problem of the bad data source. Additionally, these solutions are often too complex, and cannot be applied to certain use cases. (i.e., no "data fill" techniques can be applied to camera video streams or patient medical data).

USE CASES

When the Trust System is adopted within the business process, the initial Trust Score computation establishes a baseline or trust calibration at the very beginning of the process. During operations, as the Trust Score changes, the operator has to decide on the path forward based on other input criteria:

a) Take action to restore the Trust Score back to the original value

OR

b) Accept the newly generated Trust Score as the new normal (new baseline) by accepting the conditions that resulted in the new computation.

There may be additional options to choose from depending on the situation. In this section, we show how the Trust Score and the Trust System influence the business decisions, and also the business outcomes.

Use Case 1: Trustworthiness in a Smart, Connected Factory

Remote Factory Management

A senior team remotely manages the manufacturing of elevator parts in an overseas factory. The team is also responsible for managing several other geographically distributed factories. Cameras and sensors are used to collect data for the following purposes:

A. Identify workers at a specific workstation
B. Observe the manufacturing process at each workstation
C. Measure the production efficiency
D. Manage the safety and security of the factory floor

Collection of Data

The goals A through D above are achieved by analyzing the camera video stream with the sensor data. The cameras capturing the workstation video feed have to be available at all times as do the sensors and the network that they are connected to. Unreliable data (whether video or sensor) impedes efficient remote monitoring. Inconsistent data makes it impossible for the management to monitor the product quality or to make continuous improvements to achieve their business goals.

Workstation Setup

Figure 7: Trust computation involves each component of the workstation

Figure 7 shows an example of a workstation in this elevator parts factory. Cameras (5 or 6) mounted at each workstation capture a continuous video stream of the set of tasks related to that workstation. A worker badges in and out at the start and end of their shift at the workstation. Sensors attached to drill bits and cutting machines send data that is used to measure different aspects of the production process and the state of the instruments at each workstation.

Trust Computation for the Manufacturing Process

In this use case, each camera, sensor, application, network connection and worker schedule, among others, possesses observed values (attributes) that can be used as input into the Trust System to calculate an overall Trust Score. If any one of those metrics changes at any instant, the Trust System

recalculates the Trust Score for each workstation and those Scores can be combined to produce an integral Trust Score for the end-to-end production process.

Business goals	• Monitor and measure manufacturing quality metrics • Manage labor costs
Core Requirements	• Remotely manage factory operations (smart factory as a service) • Capture and analyze video and sensor data for the manufacturing process • Protection of the video data to comply with a variety of legal restrictions on employee video in various countries
Key Trustworthiness Characteristics	• Reliability • Security • Safety
Attributes	• Video data • Sensor data • Network connectivity • Worker identification • System uptime

Table 1: Practical application of Trustworthiness in a Smart, Connected Factory

If any single input changes (for example, a video stream is missing or sensor data is interrupted, or the wrong worker is at the workstation), this may affect not just that workstation but the entire production line and the end-to-end process. The operator decides if the change in input was due to a known (trusted) event or not. Combining this kind of environmental information with the dynamic Trust Score calculation is important for the proper functioning of the Trust Score system.

Figure 8 illustrates how the trust model measured the decrease in system trustworthiness when network connectivity was insufficient to transmit all the required data streams reliably. The manufacturing team was able to diagnose the issue based on the immediate notification provided by the decrease in the overall score supplemented by the detail of the component scores. This example also illustrates that the team has weighted the attribute of Reliability more heavily than the attributes of Security or Safety.

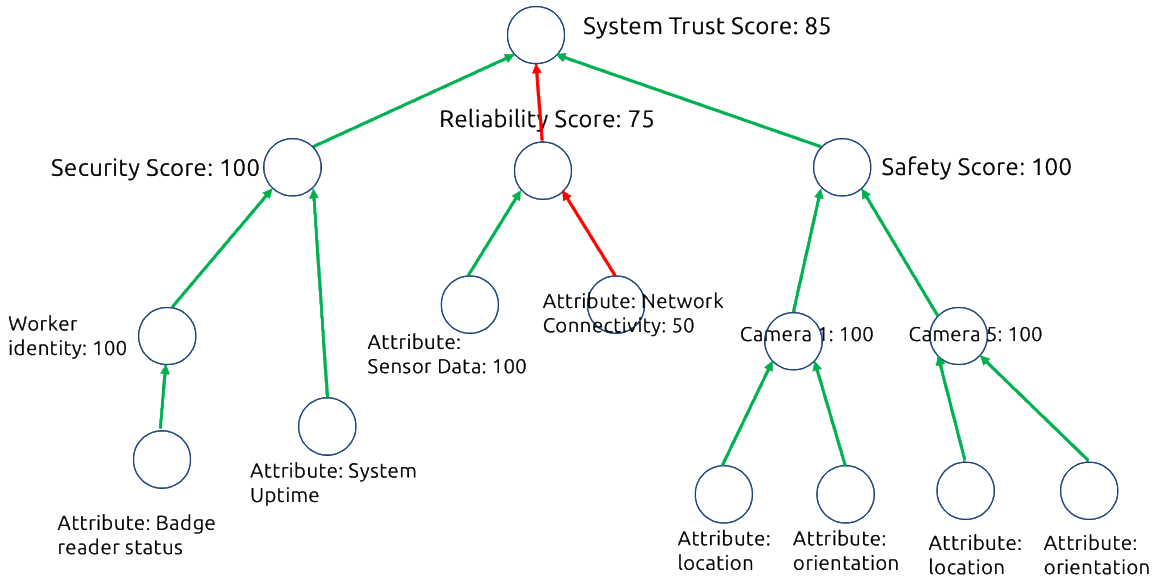

Figure 8: Computation of System Trust Score in a Smart, Connected Factory

As a result of these observed overall and component Trust Scores, the factory management team was able to modify the design of their network environment to adequately handle the connectivity requirements for transmission of video and sensor data streams. The updated network design increased the network connectivity component score, which in turn brought the reliability attribute score and the overall system Trust Score back to the desired state of 100%.

Use Case 2: Trustworthiness in a Retail Store

SMART STORE ANALYTICS

A Systems Integrator installs a surveillance system inside a retail store to provide a data collection service. Temperature and movement sensors, people counters and cameras collect data to calculate occupancy and foot traffic. The system is also used for safety and security reasons. Some important observations that retail store management is interested in are:

- How many customers entered a store at a certain time of day?
- How many customers interacted with an item on display (electronics, shoes, etc.)?
- How much time did they spend on average in a specific aisle or display?
- How many people visited a certain aisle in the store?

The data is analyzed to generate a result that is useful to achieve certain business goals:

- Are there enough sales or customer service employees to handle the inflow of customers?
- What is the level of customer service?
- Is a display attractive or informative enough to engage customers?
- Is the time spent browsing a product or in a specific aisle indicative of interest and eventual purchase?

TRUST AS A SERVICE

The system integrator provides a supplemental trust service to the management team (retail store) which includes elements of the Trust System, in order to improve the confidence in the data quality since it is collected in this dynamic, uncontrolled environment. The trust service includes a dashboard that indicates the Trust Score of the system as the data is generated and also provides historical information. The Trust System and Trust Score, included in this value service, is indicative of the confidence level of the system that is collecting the data, the collection process and the raw data itself. The raw data and the Trust Score influence the final business key performance indicators (KPIs).

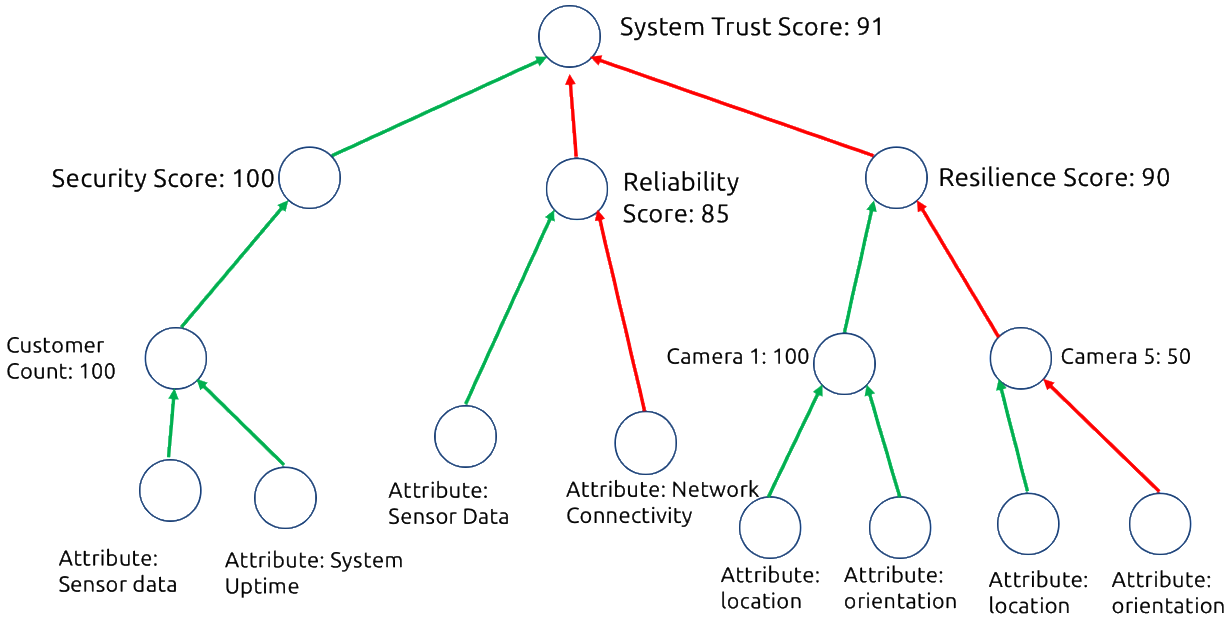

Figure 9: Computation of System Trust Score in a Retail Store

Business goals	• A Retail store proprietor incorporates smart store analytics to increase profits • A system integrator provides the system and a Trust Service to the proprietor
Core Requirements	• Capture and analyze video and sensor data for store, product marketing • Detect tampering and theft (of system and merchandise)
Key Trustworthiness Characteristics	• Security • Reliability • Resilience
Attributes	• Camera position • Camera and sensor reliability • Application software

Table 3: Practical application of Trustworthiness in a Retail Store

CONCLUSIONS AND FURTHER WORK

In this paper, we presented a practical approach for a model representation of an IIoT system's trustworthiness. The Trust Model is designed to flexibly address the varying priorities of different types of IIoT systems to provide the appropriate context for the system. The component and system Trust Scores that are computed by the Model provide an easily interpreted value that enables the model user to take the appropriate actions to maintain the trustworthiness of the system in operation. We have not shared specific details on the Trust Function algorithm or the specific visualization of the model output as these can be vendor-specific and proprietary.

The Trust System takes a set of expected and observed attributes that describe the system under observation and computes a Trust Score. The Trust Score, when combined with the analytics data, gives the observer a practical method to evaluate the system and to apply the business model with a measurable degree of confidence.

In the use cases described above, the data collected is not so unusual or unique in and of itself. However, results of the analytics are only as good as the video or sensor data that is collected. The Trust Score generated by the Trust System tackles the original problem of an unreliable data source.

There is ideally a combination of multiple cameras and sensors that provide enough coverage. What if only subsets of the cameras or sensors are operating correctly? What If there is an intermittent connectivity issue with one of the cameras? Either of these conditions will result in a lower Trust Score that is computed from the individual attributes of the system. This results in suboptimal data collection which results in suboptimal data analytics and correlation.

Ultimately, this results in mediocre business models and inadequate business KPIs due to poor data management and collection. In summary – bad data leads to poor business decisions.

Going forward, we are continuing to develop and enhance the Trust Model through further development including recording trust attributes and scores in a distributed ledger, improving the representation of component Trust Scores, potentially standardizing on a common algorithm, and extending the use cases covered by the model.

➢ Return to IIC Journal of Innovation landing page for more articles and past editions.