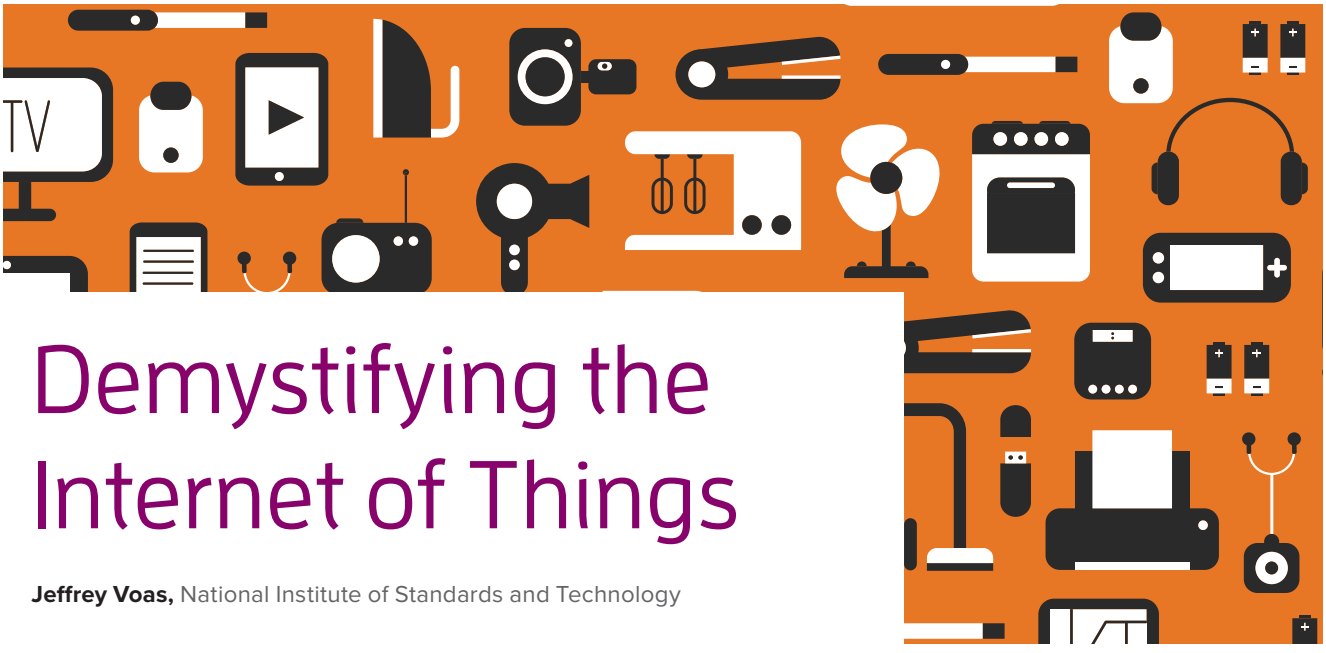

Demystifying the Internet of Things

Jeffrey Voas, National Institute of Standards and Technology

In attempting to define the Internet of Things, we should keep in mind that it's fundamentally about communication, computation, sensing, and actuation.

What is the Internet of Things (IoT)? Is it the next technology revolution? Is it a technology evolution? Is it only an initialism? Is it definable, and, if so, what's the definition? Claims about its potential impact remind me of previous "hot" technologies such as object-oriented programming, agile programming, design patterns, Y2K solutions, e-commerce, the World Wide Web, and so on.

Why do we need to define it? Two reasons: the IoT's predicted trillion-dollar economic benefits,[1] and the potential for unprecedented security and privacy risks.[2] Although these reasons offer some insight on the IoT's noteworthiness, they do nothing to inform us about what it is.

When there's a "definitional vacuum" for a new technology, many suggestions are offered. Here's a small sampling of new IoT definitions that have surfaced since May 2015:

> *The term Internet of Things generally refers to scenarios where network connectivity and computing capability extends to objects, sensors and everyday items not normally considered computers, allowing these devices to generate, exchange and consume data with minimal human intervention.*[3]

> *Although there is no single definition for the Internet of Things, competing visions agree that it relates to the integration of the physical world with the virtual world—with any object having the potential to be connected to the Internet via short-range wireless technologies, such as radio frequency identification (RFID), near field communication (NFC), or wireless sensor networks (WSNs). This merging of the physical and virtual worlds is intended to increase instrumentation, tracking, and measurement of both natural and social processes.*[4]

> *Industrial Internet of Things (IoT) is a distributed network of smart sensors that enables precise control and monitoring of complex processes over arbitrary distances.*[5]

> *The concept of Internet of Things ... is that every object in the Internet infrastructure is interconnected into a global dynamic expanding network.*[6]

> *In what's called the Internet of Things, sensors and actuators embedded in physical objects—from*

0018-9162/16/$33.00 © 2016 IEEE

EDITOR **ROY WANT**
Google; roywant@google.com

roadways to pacemakers—are linked through wired and wireless networks, often using the same Internet Protocol (IP) that connects the Internet.[7]

The Internet of Things ... refers to the trend to include networking and computing in a wide range of devices, such as watches, appliances, health monitors, toys, etc.[8]

We must first define what we mean by "things." It could be very simple objects or complex objects. Things do not need to be connected directly to the public Internet, but they must be connectable via a network (which could be a LAN, PAN, body area network, etc.). The IoT is the network of physical objects that contain embedded technology to communicate and interact with the external environment. The IoT encompasses hardware (the "things" themselves), embedded software (software running on, and enabling, the connected capabilities of the things), connectivity/communications services, and information services associated with the things (including services based on analysis of usage patterns and sensor or actuator data). An IoT solution is a product (or set of products) combined with a service either a one-to-one or a one-to-many relation. Meaning one service is combined with one (set of) product(s), or one service is combined with multiple (sets of) products.[9]

Common words among these definitions include: things, interconnected, Internet, cyber-physical, devices, sensors, smart, physical objects, and virtual world. But these definitions still don't define the IoT at a foundational level. In my search to demystify the IoT, I started with the belief that it's fundamentally about *communication*, *computation*, *sensing*, and *actuation*. Now I'll show you where that thinking led me.

FROM THE EDITOR

The Internet of Things (IoT) is like other hot technologies, in that as soon as it became a buzzword its scope and meaning seems to depend on the context—academic research, industrial applications, market development, and so on. In this month's column, NIST's Jeffrey Voas attempts to de-blur the IoT's true meaning as he proposes a more precise, constraint-friendly definition to satisfy the various groups that want to use it. –*Roy Want*

PRIMITIVES AND ELEMENTS

I first divided communication, computation, sensing, and actuation into core distributed system components termed "primitives."[10] Then I defined a class of "elements" that allow for the foreshadowing of the trustworthiness of systems built from IoT components, services, and commercial products.[10]

Before I explain primitives and elements, I should mention that I'm not satisfied with the initialism "IoT" because it's not possible to compare one IoT to another. Thus, I needed a replacement, which became the Network of Things (NoT). The difference between NoT and IoT is subtle. The IoT is an example of an NoT; more specifically, the IoT's "things" are tethered to the Internet.[11] A different type of NoT could be a local area network (LAN), with none of its "things" connected to the Internet. Social media networks, sensor networks, and the Industrial Internet are all variants of an NoT. This differentiation in terminology makes it easy to separate out use cases from varying vertical and quality domains (such as transportation, medical, financial, agricultural, safety critical, security critical, performance critical, and high assurance, to name a few). This is useful because there's no singular IoT. With this adjustment, I can now compare one NoT to another.

Primitives

Primitives are building blocks offering a unifying vocabulary that allows for composition and information exchange among differently purposed networks, such as NoTs. They provide clarity regarding more subtle concerns, including interoperability, composability, and continuously binding assets that come and go on the fly. Because no simple, actionable, and universally accepted definition for the IoT exists, the model and vocabulary proposed here reveal underlying foundations of the IoT, exposing the ingredients (or primitives) that can express how the IoT behaves without defining it.

The primitives proposed in an interagency report (IR) for the National Institute of Standards and Technology (NIST), referred to here as Draft NIST IR 8063,[10] are as follows:

- *Sensor*: an electronic utility that digitally measures physical properties (such as temperature, acceleration, weight, sound, and so on) and outputs raw data.
- *Aggregator:* a software implementation based on a mathematical function that transforms and consolidates groups of raw data into intermediate data.
- *Communication channel*: a medium by which the data is transmitted (such as physical via USB, wireless, wired, verbal, and so on) between the other primitives.
- *eUtility (external utility):* a software or hardware product or

service providing computing power that aggregators will likely not have.
- *Decision trigger:* this creates the final results needed to satisfy the purpose, specification, and requirements of a specific NoT. Decision triggers can fire actuators.

In my four-part model, the sensor handles sensing; the communication channel handles communication; and the aggregator, eUtility, and decision trigger handle computation. The decision trigger also fires up actuators if they exist, but not all NoTs will interact with actuators. Additionally, the aggregator handles NoT issues associated with big data.[12]

Elements

To complete the model, Draft NIST 8063[10] proposes six elements: environment, cost, geographic location, owner, Device_ID, and snapshot. Although not primitives, these elements play a major role in fostering the degree of trustworthiness of a purposed NoT:

- *Environment:* the universe in which all primitives in a specific NoT operate; this is essentially the operational profile of an NoT. Analogies are the various weather profiles in which an aircraft operates or a particular factory setting in which an NoT operates. Environment will likely be very difficult to define correctly.
- *Cost:* the expenses (time and money) that a specific NoT incurs in terms of nonmitigated reliability and security risks; additionally, the costs associated with each of the primitive components needed to build an NoT. Cost is an estimation or prediction and drives the design decisions in building an NoT.
- *Geographic location:* the physical place where a sensor or eUtility operates or was manufactured. Manufacturing location is a supply-chain trust issue. Note that the operating location might change over time, and that a sensor's or eUtility's geographic location along with communication channel reliability might affect the timeliness of dataflow throughout the workflow. Geographic location determination might not always be possible.
- *Owner:* the person or organization that owns a particular primitive. There can be multiple owners for any of the primitives. Note that owners could have nefarious intentions that affect overall trust, and some owners might remain anonymous.
- *Device_ID:* a unique identifier for a particular primitive. This will typically be created by the originator of the entity, but it could be modified or forged.
- *Snapshot:* an instant in time utilized for synchronization of events fired by any of the five primitives.

RELIABILITY AND SECURITY OF PRIMITIVES

As previously mentioned, primitives are the building blocks of NoTs, and elements lay out key contextual issues related to the trustworthiness of a specific NoT. Because trustworthiness is such a broad concept, my research focuses on two factors related to the five primitives: security and reliability. People often ask for simple examples of real or hypothetical use cases relating these two factors. The following are examples of simple, hypothetical reliability and security scenarios associated with each primitive.

Sensor reliability: a modern car's speed sensor is exposed to heat, water, and dust. Years later, it starts providing inconsistent readings due to this naturally occurring corruption. This is an example of malfunctions caused by environmental conditions.

Sensor security: a smart building's temperature sensors are easily accessible, and this particular system doesn't provide a means for validating the firmware's authenticity. An attacker substitutes the firmware with one that responds to remote commands. These sensors then become part of a botnet and can contribute to distributed denial-of-service (DDoS) attacks. This is an example of physical tampering and altering firmware.

Aggregator reliability: in a smart city environment, thousands of sensors transmit data to a series of smart gateways that effectively compress several gigabytes of raw data into meaningful information. A blackout that occurred in part of the city creates an unexpected condition that results in division by zero, which causes the application to keep crashing for the entire duration of the blackout. This is an example of unpredicted conditions that lead to undefined behavior and incorrect output.

Aggregator security: an attacker introduces a rogue sensor to a network that produces fake readings. These readings are passed as inputs to the aggregator function without any validation. The attacker launches a buffer overflow attack to gain root access to the entire middleware infrastructure (gateway). This is an example of an injection attack or buffer overflow.

Communication channel reliability: smart building sensors for regulating lights and temperature communicate wirelessly via IEEE 802.11 with the rest of the system. During a conference, a large number of people are gathered inside a room, having enabled Wi-Fi on their smartphones. Due to overpopulation of the channel, there are frequent disconnections and service degradation. As a result, the sensors are unable to provide readings with their predefined frequency. This is an example of loss of service due to overpopulation and connection problems.

Communication channel security: a wearable activity tracker is attached to a person's wrist and measures heart rate and blood pressure. It communicates via Bluetooth Low Energy (BLE) with the wearer's smartphone and forwards the data to his physician. Despite the fact that BLE takes specific actions to randomize the MAC address of the devices, the manufacturer neglected this feature. An attacker with a high-gain antenna can track the presence of the wearer in a crowd and create a movement profile. This is an example of eavesdropping on the communication channel.

eUtility reliability: a point-of-sale system conducting automatic smart payments depends on a cloud service for verifying the identity of the person using a card. System maintenance of the server happens to occur during business hours, which can cause delays in verification. This is an example of system failures that make the resource unavailable.

eUtility security: a smart home has a security camera installed at the front door, which sends data to a corresponding cloud application that forwards notifications and video footage to the homeowner's device when motion is detected. An attacker hires a hacking squad to conduct DDoS attacks on the application provider's servers for two hours. They're able to break into the house without the user being notified. This is an example of a DDoS attack.

Decision trigger reliability: the logic implemented is a or b versus a and b. This is an example of defective code.

Decision trigger security: the software that implements the decision trigger accepts malicious inputs, or the outputs from the trigger are sniffed and released to competitors unbeknownst to the legitimate owner of the trigger. This is an example of code tampering.

In this short article, I've offered a few arguments against taking a "one size fits all" approach to defining the IoT. Currently, the IoT initialism is closer to a marketing brand or catalog of services and products than it is to a singular, well-defined technology. Going forward, my research will start drilling down into subprimitives and subelements. For example, how should Device_ID be implemented? Is it simply RFID? And what about the communication channel? How many types of communication protocols for NoTs will there be, what are they, and what are the reliability and security challenges of each? And what about subprimitives and subelements for different vertical domains such as smart healthcare, transportation, agriculture, finance, and so on? A car's temperature sensor should certainly be different than one for a farm or an emergency room. This is only the beginning of a long discussion about what the IoT (or NoT) is all about.

ACKNOWLEDGMENTS

I thank Constantinos Kolias of George Mason University for suggestions on reliability failures and security attacks.

REFERENCES

1. D. Lund et al., "Worldwide and Regional Internet of Things (IoT) 2014-2020 Forecast: A Virtuous Cycle of Proven Value and Demand," IDC, May 2014; www.business.att.com/content/article/IoT-worldwide_regional_2014-2020-forecast.pdf.
2. "Gartner Says by 2020, More Than Half of Major New Business Processes and Systems Will Incorporate Some Element of the Internet of Things," Gartner, 14 Jan. 2016; www.gartner.com/newsroom/id/3185623.
3. K. Rose, S. Eldridge, and L. Chapin, *The Internet of Things: An Overview*, white paper, The Internet Society, Oct. 2015; www.internetsociety.org/sites/default/files/ISOC-IoT-Overview-20151221-en.pdf.
4. J. Winter, "Algorithmic Discrimination: Big Data Analytics and the Future of the Internet," *The Future Internet: Alternative Visions*, J. Winter and R. Ono, eds., Springer, 2015.
5. B.A. Huberman, "Ensuring Trust and Security in the Industrial IoT," *Ubiquity*, Jan. 2016; http://ubiquity.acm.org/article.cfm?id=2822883.
6. M.S. Farash et al., "An Efficient User Authentication and Key Agreement Scheme for Heterogeneous Wireless Sensor Network Tailored for the Internet of Things Environment," *Ad Hoc Networks*, vol. 36, no. P1, 2016.
7. M. Chui, M. Löffler, and R. Roberts, "The Internet of Things," *McKinsey Q.*, Mar. 2010; www.mckinsey.com/industries/high-tech/our-insights/the-internet-of-things.
8. B. Siever and M.P. Rogers, "A Hands-On Introduction to the Internet of Things," *Proc. 47th ACM Technical Symp. Computing Science Education* (SIGCSE 16), 2016.
9. F. Jammes, "Internet of Things in Energy Efficiency," *Ubiquity*, Feb. 2016; http://ubiquity.acm.org/article.cfm?id=2822887.
10. J. Voas, "Primitives and Elements of Internet of Things (IoT) Trustworthiness," Draft NIST IR 8063, Nat'l Inst. of Standards and Technology, Feb. 2016; http://csrc.nist.gov/publications/drafts/nistir-8063/nistir_8063_draft.pdf.
11. D. Myers, "Can You Take the Internet out of the Internet of Things?," TechCrunch.com, 5 Mar. 2016; http://techcrunch.com/2016/03/05/can-you-take-the-internet-out-of-the-internet-of-things.
12. J. Voas and C.K. Hansen, "IoT's Special Gift to Big Data," ECN, Jan. 2016; www.ecnmag.com/blog/2016/01/iots-special-gift-big-data.

JEFFREY VOAS is a computer scientist at the National Institute of Standards and Technology. Contact him at jeff.voas@nist.gov.