Lecture 20: The Internet of Things

Anirudh Sivaraman

2017/11/27

Like datacenters, the Internet of Things (IoT) is another use case for networking that has emerged over the last few years. And like datacenters, there is a history of previous technologies that were predecessors to IoT.

Let's break down the term IoT first. The term "thing"[1] in IoT refers to any device that combines some mechanism to sense the physical environment, some amount of computation, and a means of communicating the sensed (and potentially) processed data to an interested entity. One example of a "thing" is an Internet-connected video camera at a road intersection. Here, the mechanism to sense the environment is a camera, the computation is likely an embedded processor integrated into the camera, and the camera has a WiFi or cellular network interface card to connect to the Internet. The term Internet in IoT refers to the fact that these "things" are all connected to the broader Internet.

At least two attributes distinguish a "thing" from standard computers (e.g., servers, desktops, laptops, tablets, and even phones). First, "things" are made for a specific purpose and are typically constrained enough that they cannot run arbitrary applications—unlike a server/desktop/laptop that can run binaries downloaded from the Internet and a phone that can run mobile apps. Second, there is a sensory aspect to "things," which allows them to integrate communication and computation with the external environment around them. Although, mobile phones today come equipped with an array of general-purpose sensors that sense mobility (e.g., accelerometers, gyroscopes, and GPS), the sensors on "things" tend to be far more specialized to the use case on hand (e.g., a temperature and humidity sensor for a soil and/or environment monitoring application, or a high resolution camera for a security application).

1 The history of IoT: Smartdust

The immediate predecessor of IoT is a concept that was colloquially labelled *smartdust* and peaked in activity in the early 2000s. The idea was that you could sprinkle a collection of smart and small devices, then called motes, that integrated sensing, computation, and communication and could use these devices to sense the environment around you. These motes were designed to operate with little to no user intervention: a mote that was deployed in the field (say to sense soil conditions) was expected to last up to a year on a small AAA battery. There were also a few collection towers that collected data from these motes and relayed this data to the Internet. To get to these collection towers, the motes had to use each other to relay sensed data to the mote closest to the collection tower, which got the data to the collection tower and through that to the wider Internet.

This period saw the development of several networking protocols tailored to this *multi-hop* nature of sensor networks [10, 9], where data from a single sensor mote "hopped over" (was forwarded by) several different intermediate sensor motes until it made its way to the collection tower. It also saw the development of embedded operating systems (e.g., TinyOS [11]) for the constrained computational capabilities of sensor motes.

Finally, sensor networks also featured in-network aggregation techniques [13] where sensor motes aggregated data into a reduced form before relaying it to subsequent sensor motes. A simple form of aggregation is as follows: let's say the collection tower is not interested in the raw sensor readings, but rather the average of the sensor readings from as many sensor motes as possible. Then, each sensor mote could add together a sensor

[1] We'll use the term "thing" (within double quotes) to refer to an IoT device throughout this lecture to distinguish it against the common noun version of thing.

reading that it receives along with its own sensor reading before forwarding it to the next sensor mote because the addition operation is commutative and associative. The impetus behind such aggregation was the reduction in the data being forwarded over extremely low-capacity wireless links between sensor motes.

2 IoT today

The development of IoT today shares many similarities with smartdust from the decade before: in particular, the unification of sensing, communication, and computation. But, there are also several notable differences:

1. The wireless communication links for many "things" have better link capacities. While sensor motes in the past used low-energy physical layer protocols such as Zigbee, several "things" today (e.g., a security camera at home) can connect to the Internet using WiFi, just like more mainstream devices such as laptops and phones. There continue to be "things" (e.g., a Fitbit) that require a low-energy physical layer), such as Bluetooth Low Energy, but unlike the sensor motes of the past which were all characterized by low power consumption, there is a much more diverse design space today [6].

2. Many "things" no longer need batteries. A security camera can be plugged into the wall socket, which provides it with a much more abundant power supply than two AAA batteries. Again, similar to before, there continue to be low-power devices, such as watches that run on coin-sized batteries, but the important point is that the design space has broadened.

3. The sensor data is much richer, takes more space to store, more computation to process, and more link capacity to transmit. This is the result of using richer sensors such as video cameras ubiquitously (e.g., a city's police department running a network of traffic video cameras). Sifting through this data poses interesting research challenges.

There are at least four facets of IoT today that are interesting. This is my own personal opinion; there are likely other aspects of IoT that I am leaving out entirely. I'll discuss the first two in great detail. The first because it pertains to the class. The second because it dovetails well with our discussion of network security next week.

1. Networking: How do we connect IoT devices to the Internet?

2. Security: How do we make sure "things" aren't exposing private data (like when a user is present in his/her room) to unintended receipents? How do we ensure that "things" do not get compromised and turned into an attack army for DDoS attacks?

3. Interoperability: How do we make it easy for any "thing" to send data through any mobile phone or any WiFi router to any server—if it so desires? Today a particular "thing" only works with a particular mobile app on a particular platform and sometimes needs a particular hardware gateway (e.g, a USB dongle). For instance, you cannot connect an Apple Watch to an Android phone and you can only view the Fitbit's data using a specific Fitbit app. Most importantly, without the Android phone or Fitbit app, there is no way for the watch or Fitbit to get its data uploaded onto a server somewhere in the wider Internet. This is unlike a laptop/desktop today, where the TCP/IP stack provides interoperability by letting any web browser access any web site without requiring a specific browser or laptop for each site [16].

4. Data processing: How do we process the rich stream of sensor data coming at us?

3 Networking

Networking in IoT can take on many diverse forms. If there are no power or form factor or cost constraints, the simplest solution is to use a WiFi/cellular network to connect the "thing" to the Internet. This is similar to how an Internet-enabled Roku TV connects to the Internet using WiFi.

However, many "things" are small and/or limited by power constraints (e.g., a watch or Fitbit) or costs (WiFi antennas add to the cost of the device relative to lower power alternatives) ruling out the possibility of adding a WiFi or LTE chipset to the "thing". Instead, these devices use a low-power, low-range technology such as Bluetooth Low Energy to connect to a nearby phone or laptop. The phone or laptop has a specific application to view sensor readings from the device. This has the disadvantage that the "thing" cannot directly connect to the Internet, but it keeps the "things" themselves simple.

Bluetooth Low Energy handles the requirement of low-power, low-latency, low-range, and low-capacity networking of IoT. Frequently, the data rates might be higher, e.g., if the "thing" is generating video streams of a particular location. Or the range requirements might be more demanding, e.g., tracking a factory's inventory as it moves across the county in a cargo train. But, on the positive side, some of these scenarios may no longer be constrained by low power, e.g., a camera generating video streams may be plugged into a power source. Or, it might be OK to get the data once every hour as opposed to in real time. In summary, there is a design space to be explored here as well, and there are many sensible points in the design space that lead to different solutions.

We'll consider two examples to illustrate the diversity of this design space. First is a technology called LoRa [3], which achieves communication ranges of the order of a few km, but data rates that at most reach a few tens of kbit/s. However, LoRa requires between 10 and 500 mW of power [14], which might be too demanding for scenarios where the "thing" is expected to last several years with no human intervention required to replace the batteries [14]. There has been some research on enhacing LoRa to handle such ultra-low-power scenarios as well [14].

Another somewhat unorthodox example in this design space is the use of *data mules* to carry data over long distances at the cost of some increase in latency. A recent example of this is the use of drones to localize items using their RFID tags [12]. In this case, the data mule, a drone, carries sensor data read from the RFID tage back to a base station with the cost of increased latency. A more extreme example, turns the "thing" into both a sensor and a data mule. A recent example is the use of UAVs to periodically photograph a farm and return back to a base station [15]. Data mules have also been used to provide high-latency connectivity in rural areas [8]. At the other extreme, data mules provide high capacity high-latency networking for data transfers of up to 100 PB in a few weeks [1] (a few orders of magnitude faster than using a wired 1 Gbit/s transcontinental link for the same purpose).

4 Security

Another aspect of IoT that has raised concerns over the last few years is security, both the potential for exporting private information about a user and the potential for the "things" themselves to be compromised.

On the privacy front, researchers [7] have discovered that many IoT "things" reveal user-specific private information such as the user's zip code; temperature, light, or humidity readings; or even entire video streams. Many of these "things" do not encrypt their data before sending it over the network; we will discuss encrypting packet data when we talk about network security next week. The joke, courtesy security consultant David Alexander [2], is that instead of supporting plug-and-play, IoT today has degenerated to plug-and-pray.

The potential for compromise is equally severe. The Mirai botnet operated by compromising 2.5 Million devices [4] at its peak. These devices included Internet-enabled video cameras, DVRs, and routers. The botnet used these compromised devices to DDoS a few designated web sites by sending several 100 Gbit/s to a Tbit/s worth of traffic to these targeted web sites [5]. What is perhaps even more alarming is the simple methods that Mirai used to infiltrate these devices, which essentially amounted to trying out several commonly used passwords. We know this because a person claiming responsibility for Mirai dropped the source code for the botnet on github, where it is still publicly available! WARNING: Please don't try this attack at home even for fun. It's illegal and you will get into serious trouble.

References

[1] Amazon Snowmobile - Massive Exabyte-Scale Data Transfer Service. `https://aws.amazon.com/`

snowmobile/.

[2] IoT plug and pray all over again, says security consultant. http://www.computerweekly.com/news/4500278491/IOT-plug-and-pray-all-over-again-says-security-consultant.

[3] lora-alliance — Technology.

[4] McAfee Labs Threats Report. https://www.mcafee.com/us/resources/reports/rp-quarterly-threats-mar-2017.pdf.

[5] Mirai (malware). https://en.wikipedia.org/wiki/Mirai_(malware).

[6] Network Connectivity for IoT. https://6s062.github.io/6MOB/2017/materials/lec5-IOTx-WirelessNetworkConnectivity.pdf.

[7] Who Will Secure the Internet of Things? https://freedom-to-tinker.com/2016/01/19/who-will-secure-the-internet-of-things/.

[8] E. Brewer, M. Demmer, B. Du, M. Ho, M. Kam, S. Nedevschi, J. Pal, R. Patra, S. Surana, and K. Fall. The case for technology in developing regions. *Computer*, 38(6):25–38, May 2005.

[9] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pages 10 pp. vol.2–, Jan 2000.

[10] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Mobicom*, 2000.

[11] Philip Levis, Sam Madden, Joseph Polastre, Robert Szewczyk, Kamin Whitehouse, Alec Woo, David Gay, Jason Hill, Matt Welsh, Eric Brewer, David Culler, W. Weber, J. Rabaey, and E. Aarts. *TinyOS: An Operating System for Wireless Sensor Networks*. Ambient Intelligence. Springer-Verlag, 2004.

[12] Yunfei Ma, Nicholas Selby, and Fadel Adib. Drone relays for battery-free networks. In *SIGCOMM*, 2017.

[13] Samuel Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong. Tag: A tiny aggregation service for ad-hoc sensor networks. *SIGOPS Oper. Syst. Rev.*, 2002.

[14] Vamsi Talla, Mehrdad Hessar, Bryce Kellogg, Ali Najafi, Joshua R. Smith, and Shyamnath Gollakota. Lora backscatter: Enabling the vision of ubiquitous connectivity. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3), September 2017.

[15] Deepak Vasisht, Zerina Kapetanovic, Jongho Won, Xinxin Jin, Ranveer Chandra, Sudipta Sinha, Ashish Kapoor, Madhusudhan Sudarshan, and Sean Stratman. Farmbeats: An iot platform for data-driven agriculture. In *NSDI*, pages 515–529, Boston, MA, 2017. USENIX Association.

[16] Thomas Zachariah, Noah Klugman, Bradford Campbell, Joshua Adkins, Neal Jackson, and Prabal Dutta. The internet of things has a gateway problem. In *HotMobile*, 2015.