2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation

Tethys: Collecting Sensor Data Without Infrastructure or Trust

Holly Chiang*, James Hong†, Kevin Kiningham*, Laurynas Riliskis‡, Philip Levis† and Mark Horowitz*
*Electrical Engineering, Stanford University
Email: {hchiang1,kkiningh,horowitz}@stanford.edu
†Computer Science, Stanford University
Email: {hongjn,pal}@cs.stanford.edu
‡harmony.ai
Email: laurynas@harmony.ai

***Abstract*—Careful resource monitoring is necessary to understand usage patterns and set conservation goals in an institutional setting. Sensor systems provide data to measure consumption and evaluate the effectiveness of active interventions. However, deploying sensing systems can be difficult when infrastructure support is limited. This paper describes the process of designing Tethys, a wireless water flow sensor that collects data at per-fixture granularity without dependence on existing infrastructure and trusted gateways. Rather than rely on electrical infrastructure, Tethys implements energy harvesting to allow for long term deployment. To avoid dependence on existing network infrastructure, Tethys crowdsources the data collection process to residents' smartphones acting as gateways. These gateways are untrusted and unreliable, so Tethys implements end-to-end reliability and security between the sensing device and a cloud backend.**

We present initial findings from a deployment in undergraduate residential halls. Our results demonstrate that Tethys can capture meaningful patterns in shower use. For instance, visible water conservation signs are statistically correlated with shorter mean shower length ($p < 0.05$) and are a potential area for future studies.

***Index Terms*—Internet of Things, Wireless Sensor Networks, Crowdsourcing, Energy Harvesting**

I. Introduction

This paper presents Tethys, a water flow sensing system designed to measure consumption from showers in residential buildings. Tethys's main contributions are a physical design and network architecture that enable long term deployment and secure data collection without dependence on existing infrastructure or trusted gateways.

Through harvesting energy from water flow, Tethys supports extended sensor deployment without reliance on electrical infrastructure. Rather than depend on the availability of existing wireless infrastructure, Tethys uses a delay-tolerant network comprised of residents' mobile phones acting as opportunistic gateways. Crowdsourcing reduces the individual effort needed to collect data from a large, distributed population of sensors. However, the use of untrusted phones as gateway devices requires Tethys to enforce security and reliability end-to-end.

We present results from data collected from 23 sensors deployed in undergraduate residential halls over the span of two weeks. Our deployment coincided with an active water conservation campaign by residential housing management and demonstrated that prominent signs with conservation messages are correlated with a statistically significant difference in mean shower length ($p < 0.05$). We hope that Tethys will enable further studies, and allow for more informed management decisions.

II. Background and Motivation

Effective planning and implementation of water conservation policies requires an understanding the patterns and processes that drive consumption. Building level measurements can provide a baseline for analysis, but discard vital detail about the underlying generative processes and individual data points in the population. For instance, building level aggregation can reveal that within the span of a day a total of $5,000$ gallons of water were consumed, but cannot reveal if a decrease in water usage is due to residents taking shorter showers or fewer residents taking showers. Answering these questions requires data at the per-fixture level, at minute or second granularity.

Existing water meters that operate at the per fixture level assume that the user generating the data is also viewing and collecting the data; data is either displayed visually in real time ([1]–[3]) or immediately transferred to a trusted gateway, such as a smartphone ([4]–[6]). For IoT sensors deployed in homes, this gives users personal feedback. However, for the shared dormitory setting that Tethys targets, the data consumer is the residential building management. Existing devices do not support decoupling of data generation from data collection (e.g., delay in collection leads to data loss). Moreover, unlike existing designs, Tethys also requires that data collection be private and that data access be restricted so that adversaries cannot use the sensors to spy on other residents.

In summary, Tethys allows building managers and researchers to reliably observe building-wide shower use patterns at a fine granularity while preserving the anonymity of occupants.

III. Infrastructure-Free Sensor Design

Tethys is designed to allow for long deployments without reliance on electrical infrastructure. This is achieved through a

0-7695-6387-2/18/$31.00 ©2018 IEEE
DOI 10.1109/IoTDI.2018.00032

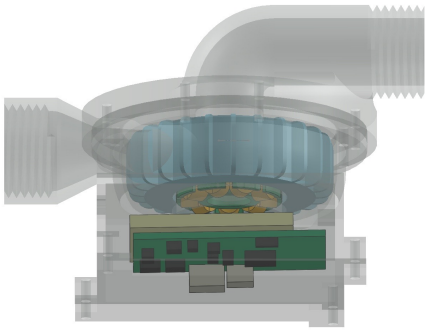

Fig. 1. Assembled Tethys sensor. The PCB (green) and battery (yellow) are placed directly underneath the stator. A thin layer of the enclosure's plastic separates the rotor and stator. Water flows in from a slit at the left, turns the rotor, and flows out to the right.

hardware and enclosure design that supports energy harvesting and a power-efficient software implementation.

A. Energy Harvesting

To power the device and recharge its battery over long deployments, Tethys uses an electromagnetic generator to harvest kinetic energy from water flow. In order to simplify the procurement of parts, we adapted an electromagnetic generator typically used to power LED shower heads for use in Tethys's enclosure. Compared to other energy harvesting approaches such as thermoelectric generation, electromagnetic energy harvesting is more invasive, requiring a unit to be inserted into the water supply. However, it can harvest nearly two orders of magnitude more power from water flow than other methods [7]. The same generator can also be used to measure flow rate by exploiting the linear correlation between the frequency of the generated voltage and flow rate.

Dormitory showers are in active use only during certain times of day and for limited time intervals. In order to support delayed data transmission when the shower is not running, Tethys requires a battery as a persistent power source. Tethys uses a rechargeable 4.2V 500mAh lithium ion battery that has high energy density in a relatively small form factor.

B. Enclosure

A custom enclosure encases the generator along with the PCB and battery. The enclosure uses an inline design; the device is mounted between a water pipe and shower head so that the generator's rotor is in line with the water flow. Figure 1 shows an assembled Tethys device.

Wireless communication necessitates an enclosure made of plastic or non-metal material so as to not interfere with radio operation. The standard approach to waterproofing consumer products is to encase the circuit in resin. However, this makes it more difficult to reprogram devices after manufacturing. Tethys instead relies on a waterproof compartment to house the PCB, which can be easily disassembled for repair or upgrades.

Manufacturing an enclosure for Tethys led to some unanticipated challenges. 3D printing enables rapid prototyping, but the choice of printing materials for actual deployment is limited. Because Tethys is an inline sensor, the material must withstand high temperature and pressure, while also being biocompatible (i.e., the surface cannot allow for buildup of bacteria or mold). Deformation in our 3D printed prototypes, led to unacceptable variation in the energy harvested. Our final enclosure uses an injection molded design, which is able to satisfy the requirements described above. Switching from 3D printing to injection molding required changes to the design; whereas a 3D print design minimizes the amount of material used, injection molding requires that a mold is simple to manufacture.

Unlike 3D printing, injection molding incurs a large initial mold cost. The mold cost for all three pieces of the design was $20,000. Economies of scale mean that at higher volumes the impact of this cost becomes negligible; at the original deployment goal of 200 devices, the tooling cost is $100 per device, bringing the total cost of a sensor to $138.36. At 2000 devices, a sensor would cost $41.86, making it competitive with commercial water sensor devices such as Amphiro A1 ($99.99) [1].

C. Embedded Device

Tethys uses a nRF51822 with a programmable Cortex M0, for processing and Bluetooth Low Energy (BLE) communication. The PCB layout is shown in Figure 2. The nRF51822 is programmed to record timestamped water flow and temperature readings. The data is encrypted on the device and sent using BLE to mobile gateways. Upon deployment, each device is initialized with the current time and encryption keys.

Water flow rate is linearly correlated to the frequency of the AC voltage generated. The generator does not spin at frequencies below 0.7 gal/min, but showers in use are unlikely to have water flow rates that fall below this amount. Tethys determines the water flow rate by sending a single phase of the generator output through the voltage clamp to an ADC pin of the nRF51822. The ADC is sampled once a second until there is a change in voltage to indicate a shower event, after which the sampling rate is changed to sample at a rate of 2ms until 250 data points are collected. The 250 data points are then used to estimate frequency by counting the number of directional transitions in voltage. The thermistor is sampled once a second.

To minimize power consumption, Tethys reduces the size of data that is stored for later transmission with delta compression; new data points are not stored unless the change in measured frequency of the generated AC waveform is at least 5 Hz or the change in temperature is at least 2°C. We tuned this threshold to effectively suppress noise.

The nRF51822 uses Nordic's S110 Softdevice implementation of BLE with an advertisement interval of 1 second. TX power is set to -8dB, which is sufficient for sensors to reliably reach smartphones in the shower rooms and hallways.

IV. Infrastructure-Free Networking

Sensors may be used in a mobile setting or in a location where existing networks inaccessible or unusable (e.g., due to

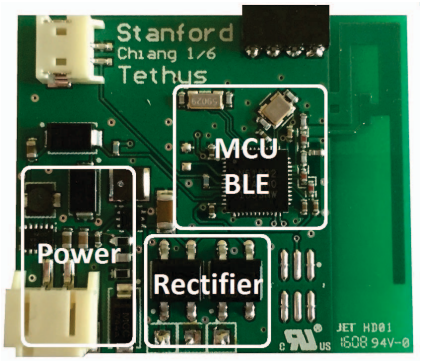

Fig. 2. Tethys PCB layout with key features labeled. Board dimensions are 33.5mm x 38.5mm x 1.6mm

power reasons). Removing a sensor's dependence on existing networks enables more flexible deployments. While Wi-Fi coverage exists in the dormitories that Tethys targets, the network is tightly controlled. Each Internet connected device must be sponsored by a permanent resident of the building, making it nontrivial or against university policy to connect sensors and custom gateways to the network.

Manually collecting data from sensors is difficult. Because devices are designed to be deployed in many buildings, it is infeasible to expect a single individual (or even a small group) to routinely service each sensor. While one option is to have sensors store data for collection at the end of the study, such a design would prevent ongoing feedback on policy effectiveness and would effectively limit the duration of the study by the size of the sensor's memory capacity.

Instead, Tethys crowdsources the data-collection task to building residents. As shown in Figure 3, the embedded Tethys sensor communicates using BLE to an app on a smartphone acting as a gateway, which in turn sends data to the cloud. The app runs in the background on users' smartphones, opportunistically collecting data when in range of a sensor. Data is stored until the phone connects to the Internet, upon which it is uploaded to the cloud servers. A resident can collect data with minimal effort; simply walking near a bathroom or periodically entering over the course of a normal day is enough to download and transmit the sensor data to the cloud. This model is suitable for Tethys as water usage results from human behavior, meaning locations that generate a lot of sensor data are also likely to be visited by gateway devices.

V. Untrusted Gateways

This section discusses the networking and security implications that result from utilizing untrusted gateways.

A. Delay Tolerant Network

Tethys makes no assumptions as to the reliability and network connectivity of individual gateway devices. Phones may not always be connected to the network while collecting data; the gateway application can be uninstalled or disabled at any minute; and data can be lost or delayed indefinitely.

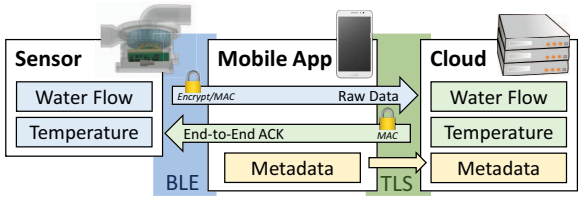

Fig. 3. Network design where the sensors transmit encrypted data packets to smartphone gateway devices which forward the packets to the cloud. The cloud provides cumulative acknowledgements back to the sensor, which stores data packets in flash until it receives an acknowledgement.

Tethys addresses these issues by implementing reliable, delay-tolerant, end-to-end acknowledgement of data from the sensor to the cloud.

Sensor readings and metadata are segmented into packets and acknowledged cumulatively by the cloud backend. A sensor must keep packets in flash and retransmit them until an acknowledgement is received. At present, each sensor retransmits data in an opportunistic manner, flooding data (starting at the first unacknowledged packet) to each phone-gateway that connects. However, Tethys's design easily allows for more conservative and less energy-intensive policies to be added in the future.

As the network is delay tolerant, a packet may arrive at the backend multiple times or out of order. These conditions are handled by the backend server.

The latency for acknowledgement delivery is reduced by periodically prefetching acknowledgements in the gateway application for all or a subset of sensors. In an average 300-person dormitory, each hall contains approximately 25 residents and 4 shower fixtures. Each acknowledgement is a 20B packet consisting of a packet sequence number and 128-bit keyed MAC. Thus, prefetching acknowledgements per hall incurs a relatively insignificant space overhead.

B. Data Security

While end-to-end acknowledgement of data prevents data loss, allowing any untrusted phone to act as a gateway device means that the design also needs to ensure that users cannot snoop packets or corrupt measured data. Such behaviors, if allowed, can have negative privacy implications and undermine the quality of the collected data. Thus, Tethys requires end-to end security as well as reliable delivery.

BLE only provides per-hop encryption and integrity, which is insufficient if the gateway is untrusted. Instead, Tethys uses end-to-end authenticated encryption between the embedded sensor and cloud server, above the BLE layer. Sensors ensure that data is encrypted and MACed before being sent to a gateway device. Acknowledgement numbers from the cloud server to the sensor are MACed but do not need to be encrypted and confidential.

The design exposes the limitations that are faced when trying to build secure applications on top of resource-constrained embedded devices. The maximum packet payload for BLE is limited by hardware support and is typically 20B for many

Fig. 4. Conservation message advocating residents to take shorter showers.

chips. This leaves little room for both data and MACs in a single packet. Using multiple packets is possible but also less efficient. Our implementation sends an acknowledgement number and a SHA256-HMAC, truncated to 128-bits (which is adequate for our deployment). The nRF51822 also only supports AES encryption (not decryption) limiting the encryption modes that can be used if both encryption and decryption are required.

Data from Tethys sensors is encrypted with Offset Codebook Mode (OCB) with AES128. OCB provides authenticated encryption, with an Encrypt-then-MAC strategy. Each Tethys device shares a unique symmetric key with the backend server. This is installed onto the device and server out-of-band. To reduce the per-packet overhead of a large 128-bit MAC for each packet, we use a block size of 128-bits and encrypt in batches of 9 packets, with every 10th packet being reserved for the MAC. One consequence of this design is that the backend server only decrypts if it knows that all 10 packets have been received and are valid. Together with authenticated acknowledgements, these measures prevent malicious or curious users from snooping, tampering with, or fabricating data and control traffic.

VI. Data Analysis and Evaluation

We analyze the data collected from an initial deployment of sensors in bathrooms across three undergraduate residence halls. The residences were chosen for uniformity of layout and design, with each bathroom containing two showers. At the time of deployment, some showers already contained water conservation messages as seen in Figure 4. Data was collected from 23 sensors over the span of 2 weeks, with a total of 1017 showers recorded. We measure how our results compare against initial hypotheses about data and water use patterns.

A. Compression Efficiency

Tethys uses delta compression to reduce the amount of data that needs to be stored and transmitted. Sensor data is only recorded if there is a change in water flow or temperature above a heuristic threshold. A compression ratio of 10 means that a 300 second shower that would create 300 datapoints without compression is compressed into 30 data points. For

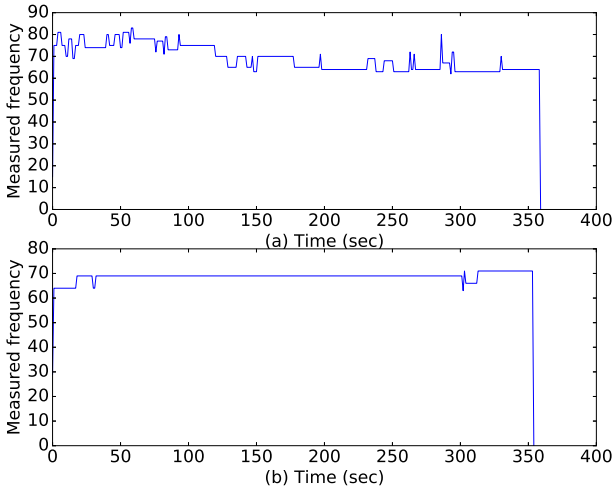

Fig. 5. Recorded shower data from a single sensor with delta compression ratio of (a) 7.0 and (b) 35.4

the task of counting and measuring shower length, a high compression ratio is desirable as it means less data needs to be stored and transmitted.

Initial predictions were that most recorded data points would be due to temperature changes at the beginning and end of a shower, as water flow is rarely adjusted once a shower is on. With a predicted average shower length of 10 minutes, we hypothesized that the average compression ratio would be 30, with around 20 data points per shower and variance caused by differences in shower length.

However, deployment results show that half the total showers have compression ratios below 20, with a degree of high variance overall. The low compression ratio and the high variance can be explained by Figure 5, where the graphs show data from two different showers recorded by the same sensor. As the fixture and sensor remain constant, differences between the two graphs must be due to variations in pressure from localized water usage: e.g., someone using the sink or flushing a toilet.

B. Showering Patterns

We used data collected by Tethys to test several hypotheses about showering behavior. The distribution of measured shower lengths is displayed in Figure 6. The median shower length is 7.77 minutes, with a surprising 27.5% of showers taking 5 minutes or less. About 4% of showers are over 20 minutes in length, making long showers potential targets of water conservation strategies. One interesting behavior we observed was a pattern of a short shower followed by another short shower less than a few minutes later (e.g., the user pauses the shower to apply soap or shampoo). To determine whether or not the two showers were actually part of a single shower instance, a heuristic was applied such that if two short showers occur within two minutes of each other or if a short shower occurs within one minute of a long shower, they are considered part of a single shower.

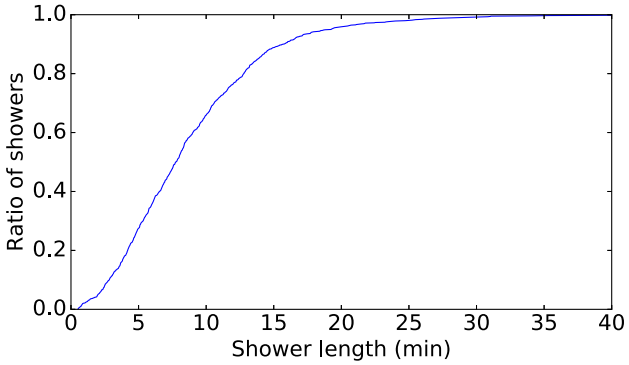

Fig. 6. Cumulative distribution graph of shower lengths.

Demographic	# sensors	# showers	Mean	Std Dev
Female	12	496	8.83	6.15
Male	11	566	8.80	5.41
Closer to door	12	577	8.57	5.66
Further from door	11	440	9.13	5.85
Has message	11	462	8.38	5.20
No message	12	555	9.18	6.15

TABLE I
IMPACT OF VARIOUS DEMOGRAPHICS ON SHOWER LENGTH: LOCATION IN FEMALE VERSUS MALE SHOWER ROOMS; LOCATION IN STALL CLOSER TO OR FURTHER AWAY FROM THE BATHROOM DOOR; WHETHER OR NOT THE SHOWER CONTAINS A CONSERVATION MESSAGE

Per fixture data allows comparison of shower patterns between demographic groups, as seen in Table I. The data shows that there is no statistically significant difference in shower lengths due to gender and that showers located closer to the door show more use. We hypothesized that the presence of a conservation message would not be sufficient to influence students' daily showing habits. However, data collected by Tethys suggests otherwise. Specifically, we observed reduced shower length in stalls with the conservation sign ($p < 0.05$). We are optimistic that Tethys will allow for more detailed evaluations of conservation policies in the long term.

C. Energy Evaluation

To be infrastructure and maintenance free, sensors must be able to operate indefinitely using energy harvested from water flow. Table II shows the most energy intensive operations performed by the device and their energy usages. In idle mode, the device has an average power consumption of 40μW. At a TX power of -8dB, each advertisement packet uses 47μJ. With an advertisement interval of 1 second, the sensor's average power consumption becomes 87μW. Assuming no energy is harvested and no data is collected or transmitted, with a 4.2V 500mAh battery the sensor can run for 2.8 years without maintenance.

The shower heads on which the sensors are deployed are designed to have a flow rate of 1 gal/min. At this rate, the sensors will harvest 15.6mW of power. To remove outlier effects, we take the median shower length, 466 seconds, and find that about 7.3J is generated per shower. At the same

Activity	Energy
Write to flash (20 bytes)	93 μJ
Advertisement	47 μJ
Connection send (20 bytes)	15 μJ

TABLE II
ENERGY USAGE IN VARIOUS OPERATION MODES.

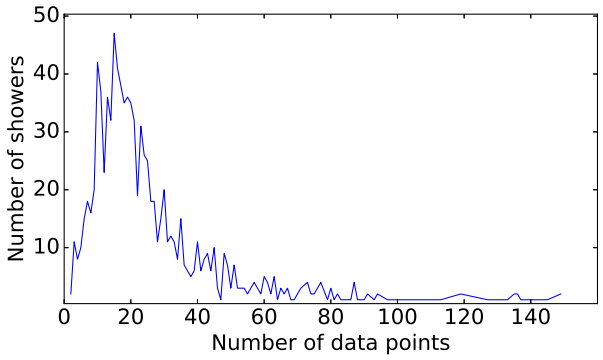

Fig. 7. Distribution of the number of datapoints contained by showers.

time, the median number of datapoints generated per shower is 21, which is a reasonable estimate as seen from Figure 7. A single shower generates around 7.8 encrypted packets. 93μJ are required to write a 20B packet to flash, and 15μJ are needed to transmit a packet over BLE. The energy cost to store and transmit the data generated by a single shower is (93 + 15)μJ/packet $\times$ 7.8 packets $\approx$ 842μJ. Given that a single shower generates 7.3J, each shower generates enough energy to retransmit its data thousands of times. Otherwise, the energy generated by a single shower is enough to extend the overall lifetime of the device by 23.3 hours.

Tethys periodically saves all state in flash so that it can continue operation after power resets. Therefore, when a sensor runs out of saved energy, once a shower does occur the sensor will be able to harvest enough energy to power on, record that shower, and transmit any new or previously stored data. Shower data will not be lost to lack of energy; data will only fail to be recorded if flash space runs out. A typical sensor records around four showers a day, which means the average sensor should be able to harvest enough power to stay on.

VII. RELATED WORK

Water Sensing: There are a variety of existing techniques for measuring water flow, including sound [2], temperature [7], and vibration [6]. Several designs use a magnetic rotor for both flow sensing and energy harvesting ([1], [3], [4]). Hydrosense takes a building-wide approach, measuring at a single point and using signal processing to identify individual fixtures and appliances. However, it is impractical for use in larger buildings as it cannot reliably identify fixtures when more than one fixture is in use [5].

Delay-Tolerant Networking: BLE is widely supported among modern phones and enables opportunistic, crowd-sourced data collection as an alternative to traditional fixed IoT

gateway designs. Past work on opportunistic wireless networks focuses on situations when connectivity is "challenged" or infrastructure is cost prohibitive ([8], [9]). [10] proposes a three-tier architecture consisting of devices, data mules, and a permanent network. [11] explores the impact of human behaviors on opportunistic IoT networking. Tethys differs from prior work in that gateways are untrusted, so that security and reliability mechanisms must be implemented end-to-end on top of delay tolerance.

Security and Privacy: Data collected by systems like Tethys can be sensitive by nature. In many settings, even seemingly innoculous data such as energy usage can be used to infer private information such as home occupancy, daily routines, and even what is on television ([12]–[14]). IoT devices are notorious for weak security due to poor design and software bugs ([15]–[17]). As we also found when designing Tethys, adding strong security features to IoT can be difficult, partially due to the challenge of implementing encryption algorithms on low power, resource constrained embedded devices [18].

VIII. Conclusion

Fine grained water usage data, at a per-fixture or individual granularity, is necessary for understanding the dynamics of communal water consumption and designing reasonable water conservation policies. Tethys addresses this need without relying on electrical infrastructure and is able to operate for months on end through the use of energy harvesting. Tethys also does not require additional gateway infrastructure, instead crowdsourcing data collection using the residents' smartphones. Finally, Tethys preserves the privacy, authenticity, and integrity of collected data despite operating over unreliable and untrusted gateway devices. Tethys has successfully been demonstrated in an initial deployment and can serve as a template for future sensing applications that need to collect data without access to infrastructure or trusted gateways.

Acknowledgment

We would like to acknowledge Noah Diffenbaugh, Sandra Nakagawa, and the Stanford R&DE staff for their helpful advice and deployment assistance. This work is supported by Intel/NSF CPS Security grant #1505684, the Secure Internet of Things Project, the Stanford Data Science Initiative, and gifts from Google, VMware, Analog Devices, and Qualcomm.

References

[1] Amphiro. (2016) amphiro. [Online]. Available: http://amphiro.com/

[2] S. Kuznetsov and E. Paulos, "Upstream: motivating water conservation with low-cost water flow sensing and persuasive displays," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10. New York, NY, USA: ACM, 2010, pp. 1851–1860. [Online]. Available: http://doi.acm.org/10.1145/1753326.1753604

[3] W. S. Hao and R. Garcia, "Development of a digital and battery-free smart flowmeter," *Energies*, vol. 7, no. 6, pp. 3695–3709, 2014.

[4] J. Leverett, Z. Wasson, and T. Lee. (2013) flow. [Online]. Available: http://jackieleverett.com/index.php/projects/flow/

[5] J. E. Froehlich, E. Larson, T. Campbell, C. Haggerty, J. Fogarty, and S. N. Patel, "HydroSense: Infrastructure-mediated Single-point Sensing of Whole-home Water Activity," in *Proceedings of the 11th International Conference on Ubiquitous Computing*, ser. UbiComp '09. New York, NY, USA: ACM, 2009, pp. 235–244. [Online]. Available: http://doi.acm.org/10.1145/1620545.1620581

[6] P. Martin, Z. Charbiwala, and M. Srivastava, "DoubleDip: Leveraging Thermoelectric Harvesting for Low Power Monitoring of Sporadic Water Use," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, ser. SenSys '12. New York, NY, USA: ACM, 2012, pp. 225–238. [Online]. Available: http://doi.acm.org/10.1145/2426656.2426679

[7] B. Campbell, B. Ghena, and P. Dutta, "Energy-harvesting Thermoelectric Sensing for Unobtrusive Water and Appliance Metering," in *Proceedings of the 2Nd International Workshop on Energy Neutral Sensing Systems*, ser. ENSsys '14. New York, NY, USA: ACM, 2014, pp. 7–12. [Online]. Available: http://doi.acm.org/10.1145/2675683.2675692

[8] M. Di Francesco, S. K. Das, and G. Anastasi, "Data collection in wireless sensor networks with mobile elements: a survey," *ACM Trans. Sen. Netw.*, vol. 8, no. 1, pp. 7:1–7:31, Aug. 2011. [Online]. Available: http://doi.acm.org/10.1145/1993042.1993049

[9] H. Nakayama, N. Ansari, A. Jamalipour, and N. Kato, "Fault-resilient Sensing in Wireless Sensor Networks," *Comput. Commun.*, vol. 30, no. 11-12, pp. 2375–2384, Sep. 2007. [Online]. Available: https://doi.org/10.1016/j.comcom.2007.04.023

[10] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: modeling and analysis of a three-tier architecture for sparse sensor networks," *Ad Hoc Networks*, vol. 1, no. 23, pp. 215 – 233, 2003, sensor Network Protocols and Applications. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870503000039

[11] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou, "Opportunistic IoT: exploring the harmonious interaction between human and the Internet of things," *Journal of Network and Computer Applications*, vol. 36, no. 6, pp. 1531 – 1539, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804513000052

[12] D. Perera, "Smart grid powers up privacy worries," 2015. [Online]. Available: http://www.politico.com/story/2015/01/energy-electricity-data-use-113901

[13] S. McLaughlin, P. McDaniel, and W. Aiello, "Protecting Consumer Privacy from Electric Load Monitoring," in *Proceedings of the 18th ACM Conference on Computer and Communications Security*, ser. CCS '11. New York, NY, USA: ACM, 2011, pp. 87–98. [Online]. Available: http://doi.acm.org/10.1145/2046707.2046720

[14] W. Yang, N. Li, Y. Qi, W. Qardaji, S. McLaughlin, and P. McDaniel, "Minimizing Private Data Disclosures in the Smart Grid," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, ser. CCS '12. New York, NY, USA: ACM, 2012, pp. 415–427. [Online]. Available: http://doi.acm.org/10.1145/2382196.2382242

[15] I. Rouf, H. Mustafa, M. Xu, W. Xu, R. Miller, and M. Gruteser, "Neighborhood watch: Security and privacy analysis of automatic meter reading systems," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, ser. CCS '12. New York, NY, USA: ACM, 2012, pp. 462–473. [Online]. Available: http://doi.acm.org/10.1145/2382196.2382246

[16] I. Rouf, R. Miller, H. Mustafa, T. Taylor, S. Oh, W. Xu, M. Gruteser, W. Trappe, and I. Seskar, "Security and Privacy Vulnerabilities of In-car Wireless Networks: A Tire Pressure Monitoring System Case Study," in *Proceedings of the 19th USENIX Conference on Security*, ser. USENIX Security'10. Berkeley, CA, USA: USENIX Association, 2010, pp. 21–21. [Online]. Available: http://dl.acm.org/citation.cfm?id=1929820.1929848

[17] E. Ronen, A. Shamir, A. O. Weingarten, and C. OFlynn, "IoT Goes Nuclear: Creating a ZigBee Chain Reaction," in *2017 IEEE Symposium on Security and Privacy (SP)*, May 2017, pp. 195–212.

[18] W. Trappe, R. Howard, and R. S. Moore, "Low-Energy Security: Limits and Opportunities in the Internet of Things," *IEEE Security Privacy*, vol. 13, no. 1, pp. 14–21, Jan 2015.