Cloud Assisted Home Networks

Hassan Habibi Gharakheili
University of New South Wales
Sydney, Australia
h.habibi@unsw.edu.au

Vijay Sivaraman
University of New South Wales
Sydney, Australia
vijay@unsw.edu.au

ABSTRACT

Managed services for the home have traditionally been shunned by Internet Service Providers (ISPs) as having high overheads and low margins. In this paper we argue that the maturing ecosystem around Software Defined Networking (SDN) changes the equation, allowing ISPs to provide cloud-based and self-managed value-add services to consumers at low cost and large scale. We first demonstrate use-cases in which SDN gives greater visibility into home network activity, enabling self-customization of Internet experience by the household, while reducing support costs for the ISP. We then outline a cloud-assisted architecture that realizes these capabilities, and detail our implementation that leverages commodity hardware and open-source software. Finally, we deploy our fully-functional system in selected households in Australia and Iran, and analyze activity data collected over a month to present insights on number and composition of connected devices, video-viewing patterns, and content preferences based on web-sites visited.

ACM Reference Format:
Hassan Habibi Gharakheili and Vijay Sivaraman. 2017. Cloud Assisted Home Networks. In *CAN '17: Cloud-Assisted Networking Workshop, December 12, 2017, Incheon, Republic of Korea.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3155921.3158432

1 INTRODUCTION

Broadband Internet service is experiencing a strange paradox: on the one hand the number of Internet-connected consumer devices (such as tablets, smart-phones, smart-TVs, gaming consoles, household appliances, and medical devices) is growing globally at an annual compound rate of 8.8% [3]; on the other hand, revenues for ISPs worldwide from fixed-line broadband are relatively stagnant [2, 9], indicating that they are not tapping into the revenue potential of device-rich household networks. While some of the barriers to the economic dividends are to do with regulation of traffic prioritization (see for example our recent survey on network neutrality [6]), there are several other economic opportunities in value-add services (related to quota management, parental controls, and security, as detailed in the next section) that ISPs are well-positioned to provide, and yet do not do so.

ISPs today avoid, rather than embrace, device-level visibility into the house, and typically deem their service to terminate at the external-facing port of the home gateway or router. This unburdens them from dealing with misbehavior or misconfiguration in (a heterogeneous set of) home devices (including the home gateway itself), which would otherwise consume excessive effort to support. From the ISP's point-of-view, "managed home services" are simply not lucrative, particularly since the residential broadband market is very competitive and hence has low profit margins.

The above approach by ISPs leaves consumers high-and-dry, as they face new pain-points arising from the increasing number of household devices sharing the residential broadband link. Many countries in the developing world limit the monthly quota on Internet data volume consumption, of which an unfair share may get used (advertently or inadvertently) by one or more household members/devices; these often lead to support calls by the consumer questioning how the data volume was used, which the ISP is unable to answer as they have no visibility of culpable device(s). In the developed world, parents are struggling to keep their kids safe from objectionable content over the Internet, or indeed to limit youth Internet addiction, but lack tools to do so. Security, particularly for connected appliances that are pervading smart-homes, remains a serious concern that is under-addressed by appliance manufacturers, and is screaming for solutions at the network-level that are lacking today.

One might be tempted to embed solutions for the above problems into home routers/gateways. We reject such an approach for multiple reasons: (a) home routers compete on price, limiting the software expertise available to vendors to embed advanced features; (b) consumers find embedded features hard to use, as corroborated by several HCI studies [10]; and (c) embedded features get outdated rapidly, and are very hard to update once deployed. We therefore believe that the ISP is best positioned to serve the above needs, and SDN provides the necessary means to do so cost-effectively and at-scale, as argued next.

The decoupling of the data and control planes under the SDN paradigm allows an elegant approach to address the above needs: the home gateway forwards data packets using match-action rules compliant with the OpenFlow standard, while the control logic that implements the features above (and appropriately configures the rule-table entries in the home gateway) resides in the cloud. This simple architecture allows any commodity home router, flashed with any OpenFlow agent (e.g. OpenVSwitch), to be used, bringing with it two significant advantages: (a) the ISP need not provide custom hardware or firmware, and can thus keep costs low by leveraging the competitive marketplace for home routers, and (b) new features can be enabled scalably from the cloud, without requiring any modifications or upgrades to the home gateway device once deployed. These aspects, as we will explain in greater depth through the course of this paper, provide a compelling business case in support of ISPs providing value-add services to consumers that have hitherto not been feasible.

CAN'17, December 12, 2017, Incheon, Republic of Korea
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-5423-3/17/12…$15.00
https://doi.org/10.1145/3155921.3158432

2 SDN USE-CASES FOR THE HOME

We briefly describe three use-cases, two of them being current and the third likely to gain prominence in coming years, of SDN-based value-add services that can be provided by ISPs to benefit consumers.

Quota management: Our first use-case is from Iran, a developing country of over 80 million people in Asia where demand for Internet access is growing rapidly. However, due to political and economic reasons, international connectivity remains slow and expensive. Consequently, typical broadband plans impose tight quotas, of the order of 5 GB per-month. When this quota is reached, the consumer is disconnected from the Internet, and they have to call their ISP to purchase additional quota (much of the developing world operates on pre-payment rather than post-pay billing). This is not only frustrating for consumers, but also imposes a heavy support cost on ISPs – our partner ISP Asre-Telecom [1], among the top-5 largest in Iran, reveals that in the last quarter of 2016, 28% of the support calls they received from their subscribers in the Tehran region were disputes related to quota exhaustion. Further, these subscribers often contest their usage, leading to complaints and disputes. It turns out that more often that not, kids in the household stream videos and photos with abandon on their smartphones/tablets, leading to quota depletion that their parents are not aware of. The remedy for such situations can be as simple as giving the subscriber visibility into Internet consumption of each device in their house, so they can at least identify culpable devices, and if desired impose a limit on specific devices so that they do not consume the entire quota leading to disconnection of the household. The ISP, by employing SDN out to the home gateway, can easily provide this visibility and control, as we will demonstrate later in this paper.

Parental controls: Youth Internet addiction is becoming a serious concern in many Western countries, and is starting to afflict the developing world as well. Research [4] has shown that children as young as ten are exposed to content containing bad language, violence, or pornography, and a worrying 42% of ten-year-olds admit to hiding their online activity from parents. Though a plethora of client-side tools are available to shield kids from inappropriate content, including child-safe DNS resolvers, search engines, browser filters, operating system modes, and free/paid software suites, their uptake is poor as they demand high motivation from the parent to install, configure and maintain. The ISP has an opportunity to fill this gap, by providing the consumer with a single tool for managing content filtering across all household devices. The use of SDN empowers the subscriber to customize (via match-action rules in the home gateway) monitoring/filtering appropriate to the composition of people and devices in their household, while allowing the ISP to constantly update (via software in the cloud) ratings for new content as it emerges.

Security for smart-homes: The phenomenon of "smart-homes" is still in its early days, but indications are that within the next few years, many Western households will be equipped with Internet-connected appliances such as lights, door-locks, smoke-alarms, and health/fitness monitors. Cisco VNI predicts that the "Internet-of-Things" (IoT) connections will grow by 34% each year, rising from 780 million globally in 2016 to 3.3 billion by 2021. Many researchers, including us, have revealed the ease with which smart-home appliances are vulnerable to hackers [5, 7], yielding private information that can be abused, ceding control to unauthorized entities, or acting as launchpads for staging large-scale cyber-attacks. Though there are several security initiatives led by large companies like Google, Apple and Samsung, we believe that the sheer volume and heterogeneity in IoT devices requires security to be addressed at the network-level, something that the ISP is well-poised to provide. Further, the rapidly evolving nature of security threats makes SDN the ideal paradigm for executing the detection and mitigation algorithms under centralized logic in the cloud.

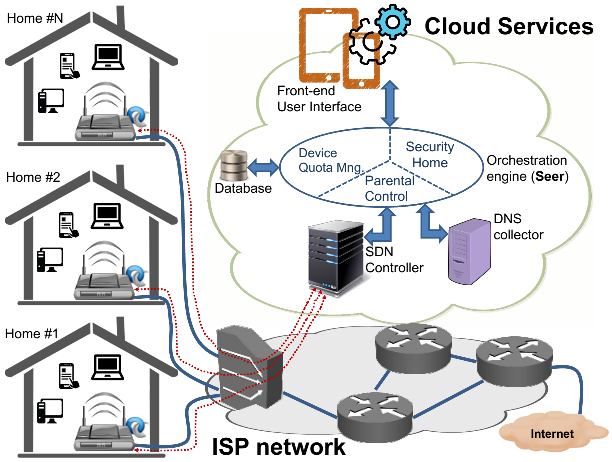


Figure 1: System architecture

3 SDN-BASED SYSTEM ARCHITECTURE

We now argue that SDN provides the best means for ISPs to offer the above capabilities to end-users, both from a technology and business perspective. Our SDN-based architecture as depicted in Fig. 1. Data flow is shown by solid blue lines, and is conventional, i.e. from the home gateway to the DSLAM and through the ISP network out to the Internet. Control flow, which is what implements the new capabilities above, is depicted via dotted red lines. An SDN controller manages the home gateways, and applications providing the features above reside on the controller and are exposed to users via a web-interface. The various components are described and justified below:

Home router: It is very tempting to embed the above features into the home gateway itself – as argued earlier, we believe this is not the right approach. ISPs rarely design or manufacture home routers themselves, and instead partner with one or more vendor suppliers. Embedding custom-built features into the home router would lock the ISP with a supplier, increasing risk and cost if the supplier under-performs or over-charges. We therefore strongly believe that the user-premises equipment should be an off-the-shelf device, that can be controlled externally using a standard well-understood interface. This is exactly what SDN provides. We have successfully taken commodity home gateways and flashed them with OpenWRT and OpenVSwitch, allowing them to function as generic switches whose forwarding behavior can be manipulated by an external controller. We emphasize that we do not customize the home gateway in any way whatsoever; this approach avoids vendor lock-in, and leverages open-source firmware (OpenWRT and

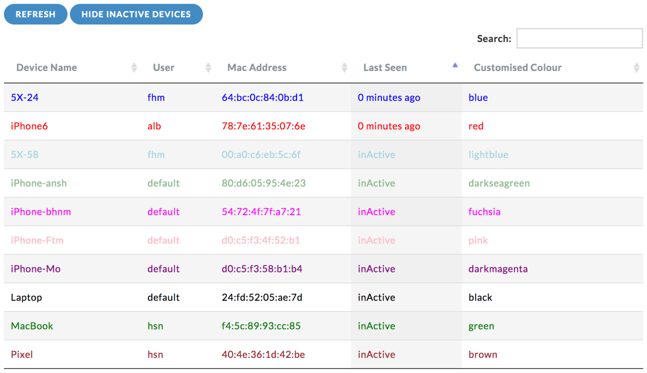
OVS) that is well-supported in the community. Equally importantly, all the control logic now resides in the cloud, allowing continual updates and customization of features and user-interfaces without requiring any modifications to the home gateways.

Controller: The controller provides the substrate on which SDN applications reside and interact with the network devices (home gateways). They provide functionality such as maintaining the list of connected devices, inserting/deleting flow-table entries in switches using the OpenFlow protocol, and polling statistics. Several open-source controllers are available for this purpose, any of which can be used. We have chosen to work with FloodLight, but have written wrappers for its functions so that our application code is agnostic to the controller, allowing us to replace the controller at any point.

DNS collector: The DNS collector receives a copy of all DNS queries generated by devices whose parental control feature has been enabled by the user. DNS packets of user-enabled devices are mirrored to the collector using specific flow entries. The collector then extracts the domain name followed by obtaining a content category type (e.g. video sharing or social networking) for the requested domain name from OpenDNS [8], and writes the domain name, tag and the device MAC address along with its time-stamp to the database.

Orchestration engine and database: The bulk of our intellectual property lies in the orchestration engine, which we call *Seer*. Seer implements the logic for the quota management, parental controls, and security functions mentioned above, along with the databases of subscribers, their devices, preferences, usage, etc. It maintains the association between subscribers and their home gateway's data-path ID, so that control commands can be sent to the appropriate device. Its interactions with the controller and the user-interface are via REST interfaces, so that the implementations of the various components is decoupled (e.g. one SDN controller can be replaced for another, and the web-based user-interface replaced by a mobile-app). The features themselves are implemented entirely in software operating in the control plane: for example quota management periodically polls the per-device counters from the home gateway to determine instantaneous and cumulative data usage, parental controls mirror DNS look-ups to record and tag visited sites depending on device profiles, and smart-home security monitors destinations of IP traffic emanating from each household appliance. We emphasize that the features are entirely implemented in the cloud and do not require any special support from the home gateway other than the standard match-action rules mandated by OpenFlow – this allows features to be added, updated, or customized by the ISP at any time.

User interface: The decoupling of features from the hardware platform permits arbitrary interfaces to be built: our current interface is web-based and interacts with the orchestration engine above using REST APIs; in the near future we intend to develop mobile apps that give a different experience for users but use the same underlying REST APIs. Yet again, we emphasize that the user-interface is in the cloud and not embedded into the home gateway (e.g. 192.168.1.1), so can be updated, improved, and customized at-will by the ISP.



Device Name	User	Mac Address	Last Seen	Customised Colour
5X-24	fhm	64:bc:0c:84:0b:d1	0 minutes ago	blue
iPhone6	alb	78:7e:61:35:07:6e	0 minutes ago	red
5X-58	fhm	00:a0:c6:eb:5c:6f	inActive	lightblue
iPhone-ansh	default	80:d6:05:95:4e:23	inActive	darkseagreen
iPhone-bhnm	default	54:72:4f:7f:a7:21	inActive	fuchsia
iPhone-Ftm	default	d0:c5:f3:4f:52:b1	inActive	pink
iPhone-Mo	default	d0:c5:f3:58:b1:b4	inActive	darkmagenta
Laptop	default	24:fd:52:05:ae:7d	inActive	black
MacBook	hsn	f4:5c:89:93:cc:85	inActive	green
Pixel	hsn	40:4e:36:1d:42:be	inActive	brown

Figure 2: Subscriber's household devices, showing name, user, MAC address, last-seen time, and colour

4 IMPLEMENTATION AND EVALUATION

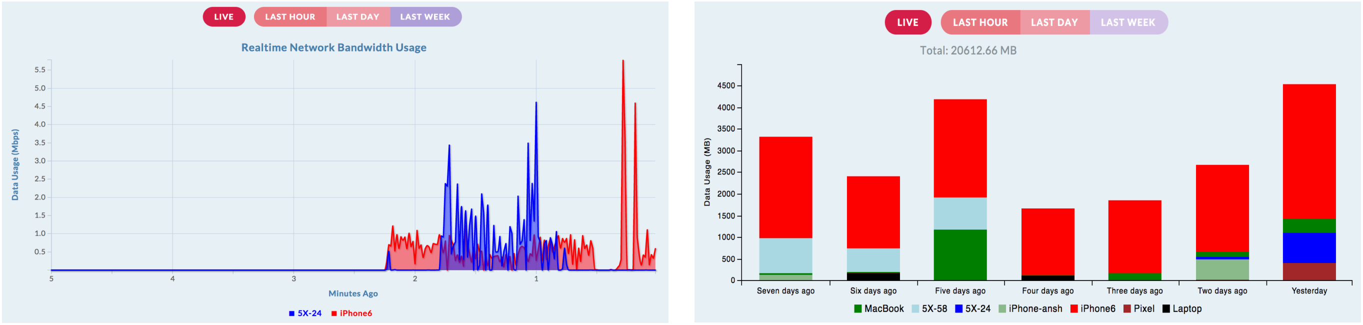
We are developing commercial-grade software that embodies the architecture depicted in Fig. 1, and are currently piloting our system with a mid-sized ISP in Iran. In what follows we briefly describe the main aspects of our implementation, and preliminary observations from the evaluations we have conducted so far.

4.1 Implementation

Home Gateway: We have tried multiple off-the-shelf home gateways, and for our pilot have chosen to use the TP-LINK Archer C7 AC1750 gateway that has 4 LAN ports and WLAN supporting 802.11ac. To minimize risk during the testing phase, we connect this gateway to the existing home gateway via the WAN port, so that the household can fail-over to its legacy network if needed. We installed OpenWrt firmware (v14.07) and OVS (v2.3.0) on our gateway, and configured an appropriate virtual bridge that takes OpenFlow commands from our controller in the cloud (described next). Our configuration scripts are open, and no custom code was written for the gateway, allowing any brand of home gateway supporting OVS to be used.

SDN controller: We used the Floodlight (v1.2) OpenFlow controller, running in the Amazon cloud, for managing the home gateways. We could equally have chosen any controller in the market, and our choice was driven by feature-set availability, maturity of the code-base, and our own comfort with its architecture. To keep our orchestrator (Seer) module agnostic to the SDN controller, we wrote a shim layer (in Python) that translates application-level APIs into the appropriate functionality in the controller. These north-bound APIs are RESTful, and provide the following functionality:

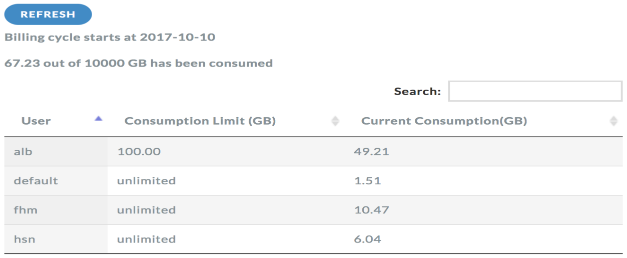
- *Device discovery:* This allows the orchestrator to query the network controller for a list of devices (identified by MAC address) belonging to a subscriber (i.e. connected to a specific home gateway).
- *Device presence:* This returns the last time at which a user device was seen in its home network. This can not only show connectivity status to the user, but also lets the orchestrator application (re)construct context information potentially relevant to security.
- *Usage statistics:* This returns downstream and upstream byte-counts pertaining to a specific device for a subscriber, and is built upon the SDN controller's ability to query for flow-level statistics from the switch.



(a) Live Usage

(b) Weekly Usage

Figure 3: Web interface showing usage: (a) live (b) weekly.



Billing cycle starts at 2017-10-10

67.23 out of 10000 GB has been consumed

Search:

User	Consumption Limit (GB)	Current Consumption(GB)
alb	100.00	49.21
default	unlimited	1.51
fhm	unlimited	10.47
hsn	unlimited	6.04

Figure 4: Quota management, showing user, quota, and total consumption

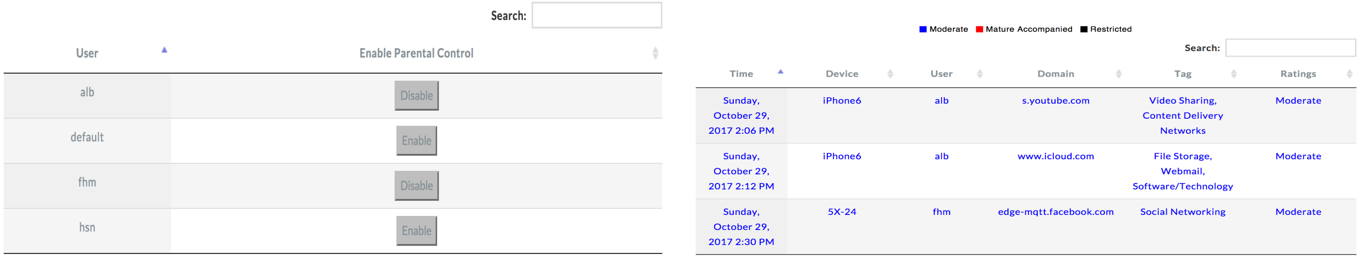
- *DNS mirroring:* This function allows all DNS requests (destination UDP port 53) from a specified subscriber device to be mirrored, so that appropriate parental controls and security management can be implemented by the application. We translate this to appropriate flow-level rules in the home gateway's OVS tables, such that it is transparent to end-user client devices and cannot be bypassed by changing their DNS settings.
- *Access control:* This allows the orchestrator application to allow or deny all traffic between a specified subscriber device and a remote IP address block, and acts as a wrapper to FloodLight's existing firewall module functionality

Seer: The orchestration engine, termed *Seer*, holds the main logic and associated databases for delivering the services described in §2. It consumes the north-bound APIs exposed by the shim layer wrapping the controller (as described above), and in turn exposes REST APIs for interaction with user-interfaces (described next). The Seer module is written in Ruby-on-Rails, and uses a PostgreSQL database with tables for subscribers, devices, policies, user preferences and statistics. The subscriber table helps map a user to the appropriate OVS DPID, so that any queries or actions invoked by the user are mapped to the correct home gateway. The devices table keeps track of the client devices connected to a subscriber's home gateway, including their MAC address, host name, last time seen, etc. The set of features supported by our platform is relatively limited (quota management, parental controls, and security), and hence we use custom-built logic based on our relational databases; in future work we intend to use formal methods for policy specification and conflict management. For concurrency to handle multiple users, we use the "sidekiq" library, and for concurrent flexible authentication we use the "devise" library in Ruby on Rails.

Web-based portal: We have built a front-end (live at `http://www.networkseer.com/`) for users to configure and customize their services, using the ReactJS Javascript library developed by Facebook. Upon first visiting the portal, the subscriber has to register and specify their ISP and customer id, so we can validate with the ISP and map the subscriber to the appropriate DPID of their home gateway. This activates the service, and the subscriber can then see their household devices via the portal. In Fig. 2 we show a sample illustration of the portal showing 10 devices for this trial user. The device name is auto-detected based on the DHCP host-name, but can be edited by the user via the interface. The table also shows the time the device was last seen on the home network, and provides capabilities to sort entries by any column, or to search through them. These greatly enhance the user experience, which can be further customized at any time by simply updating the portal software hosted in the cloud.

The portal has a tab that shows ***usage*** statistics, as depicted in Fig. 3. Clicking on the "Live" button gives the user an instantaneous view of bandwidth usage of each active device in their network over the past few minutes, as depicted in Fig. 3(a). This is useful when application quality (e.g. live teleconferencing or streaming video) suffers and the subscriber wants to check if other devices in the household are competing for bandwidth on the broadband link. The interface can also show aggregate data volume (download plus upload) for each device over the past hour, day, or week, using a stacked bar-graph, as depicted in Fig. 3(b). Our tool allows a subscriber to impose a quota on a per-user basis to pre-empt the household quota depletion problem, as shown in Fig. 4 – note that all users have "unlimited" quota by default. When the total usage from all devices associated with a user reaches to the specified quota, a blocking rule is pushed into the switch corresponding to each device of that user.

The portal has a tab for ***parental control***, as shown in Fig. 5. This allows DNS monitoring to be applied to each specific user in the household as depicted in Fig. 5(a). Upon insertion of per-device DNS mirroring rules, a detailed table of report is visualized (shown in Fig. 5(b)) live as well as for the past hour, day, and week. Currently we use domain tags (the second right column in Fig. 5(b)) provided by OpenDNS community, and rate them by five values namely "Restricted" (e.g. nudity), "Matured Accompanied" (e.g. alcohol), "Moderate" (e.g. social networking), "Parental Guidance" (e.g. web spam), and "General" (e.g. search engines).



Search:

User	Enable Parental Control
alb	Disable
default	Enable
fhm	Disable
hsn	Enable

(a) Config.

■ Moderate ■ Mature Accompanied ■ Restricted

Search:

Time	Device	User	Domain	Tag	Ratings
Sunday, October 29, 2017 2:06 PM	iPhone6	alb	s.youtube.com	Video Sharing, Content Delivery Networks	Moderate
Sunday, October 29, 2017 2:12 PM	iPhone6	alb	www.icloud.com	File Storage, Webmail, Software/Technology	Moderate
Sunday, October 29, 2017 2:30 PM	5X-24	fhm	edge-mqtt.facebook.com	Social Networking	Moderate

(b) Report.

Figure 5: Parental control, showing: (a) configure user, (b) report: time, device name, user, domain name, tag, and content rating.

The ***security*** features for protection of smart-homes is still under development, and allows the user to delegate security/privacy of any of their smart-home IoT devices to the ISP; the ISP holds the knowledge base on appropriate methods to protect that specific device, and can insert appropriate access control rules via the network API, potentially using context information from the home.

4.2 Deployment and Evaluation

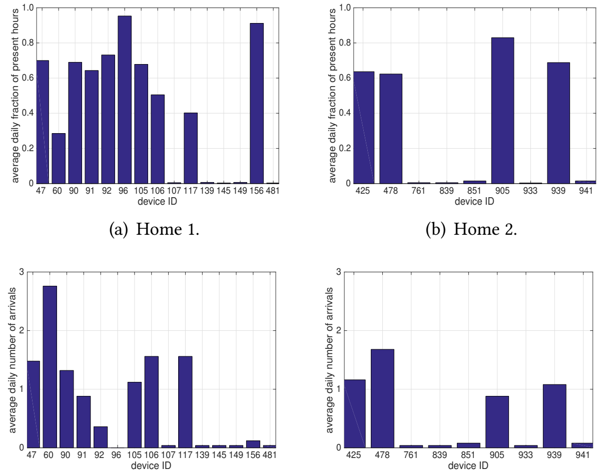
We have installed the product in a few homes in Australia and Iran, and validated and reformed its functionality and performance over the past twelve months. Early experiments showed that users were quite excited to be able to see their household devices automatically discovered and listed. Initial identification of devices by their MAC address is somewhat cumbersome, but the DHCP name provided reasonable hints (e.g. “AppleTV” or “android-phone”); the user can thereafter change the device name on the portal to make it easier for them to identify. Also, our system is fairly responsive to connection of new devices - it takes no more than a few seconds to detect and display a newly connected device; on the other hand, disconnection of the device takes over a minute to trigger a change in the user-interface, since it is detected from a periodic active scan done by the SDN controller.

The live statistics gets updated every 5 seconds, and users were excited to be able to see bandwidth usage of their devices in real-time. There are occasional spikes in the bandwidth usage graphs, which upon investigation seemed to arise from the jitter in response to the asynchronous polling of the flow-table counters. The overheads of gathering statistics is relatively low; each flow-table entry is polled every 5 seconds for live statistics and every minute for recorded statistics, and each poll is only a few hundred bytes, which comes to a few Kbps that we deem negligible. The ability to see per-device statistics is currently being evaluated for a pilot deployment to 100 customers by our partner ISP in Iran, who is facing customer complaints from subscribers who use their monthly quota but cannot identify the culpable device(s) in their house.

4.3 Analytics and Insights

We now show the benefit of our system to ISPs drawing insights into home networks. We have analyzed data from two homes that were collected for a duration of 4 weeks, from May 1, 2017 to May 28, 2017. In order to maintain privacy of users, we call the two households “Home 1" and “Home 2".

Device Composition: We first aim to identify the composition of devices in the home – knowing if a given device belongs to a person in the household (e.g. personal phone or laptop), or is shared among the family (i.e. TV or printer), or belongs to a visitor. We plot in Fig. 6 the average daily fraction of “present” hours (one minute of activity for a device causes its status to be present in that hour), and the average daily number of “arrivals” for each device in the two households (when a device status becomes present after an hour of inactivity, it implies that the device has arrived to the network). We expect devices which are always connected to the network, i.e. permanent and non-portable devices (generally shared by the family), to have a higher fraction of daily present hours. For example, devices 96 and 156 in Fig. 6(a) are usually present and active in the home. These devices correspond to TV and printer in the Home-1 respectively. While considering Home-2 in Fig. 6(b), we observe that household devices are not present for at least 4 hours a day (i.e. 18% of time). Note that devices with low fraction of present hours (i.e. close to 0) correspond to visitor devices (for example, device 481 in Home-1 and device 941 in Home-2) – this was confirmed by individual households in our study.



(a) Home 1.

(b) Home 2.

(c) Home 1.

(d) Home 2.

Figure 6: Profile of daily presence for home devices.

Looking into arrival pattern, we expect personal devices of each household to arrive to the network on a daily basis. We see that six devices of Home-1, and three devices of Home-2 arrive to their network at least once everyday, as shown in Figures 6(c) and 6(d). Unsurprisingly, permanent and visitor devices in all households have low average daily arrival values.

Usage Pattern: We then analyze usage data to identify devices which consume high volume of data and the hours in which they do so. Devices that are connected to the network but not interacting

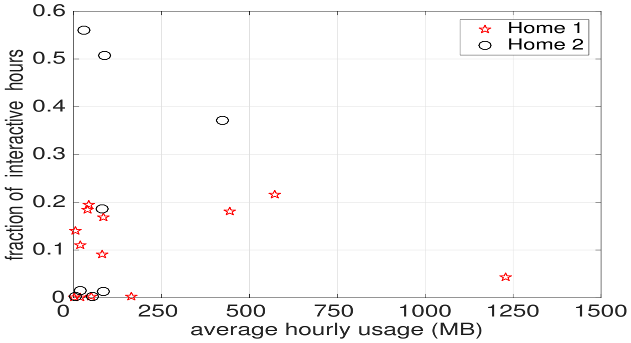


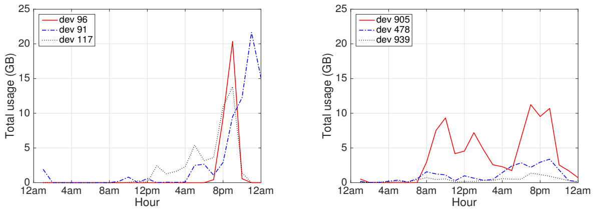
Figure 7: Average hourly usage vs. interactive hours

much with a user (e.g. a printer) autonomously generate low volume traffic. In our analysis, we deem a device to be "interactive" over an hour, if it exchanges more than 1 MB data. Fig. 7 shows a scatter plot of usage versus interactive hours for households devices. It is seen that there are four data-hungry devices in the two homes, consuming a fairly large volume of hourly data on an average (i.e more than 250 MB). This implies that three devices in Home-1 and one device in Home-2 are used for watching videos frequently. Unlike other data hungry devices, the one in Home-2 (shown by black circle) seems to interact with users more often.

We choose three data-hungry devices from each home network. In Fig. 8, we plot the total monthly volume of usage in each hour consumed by these data-hungry devices. We see a prominent pattern of high data usage, indicating video viewing, in Home-1 as shown in Fig. 8(a). Devices 96 and 117 are likely to consume videos between 9-10pm, whereas device 91 tends to play videos one hour later, i.e. between 10-11pm. Similarly, in Home-2 (Fig. 8(b)), device 905 seemingly consumes video contents at various hours of day whereas device 478 is used for watching videos during evening hours.

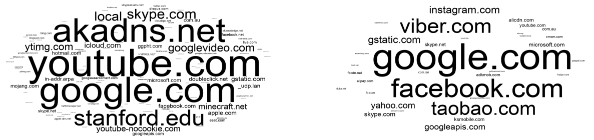
Content Preference: Finally, we aim to profile devices based on their visited domains. We note that visiting a domain sometimes generate a number of subsequent DNS queries depending on accessed content. For example, visiting Facebook or Youtube generates tens of requests to dependent or advertisement domains. These subsequent queries (i.e."background" domains) are automatically generated while loading the page, but are not user initiated. We therefore filter them in our analysis – 33% and 48% of DNS queries contribute to background domains in homes 1 and 2 respectively.

We select a personal device from each household and try to infer the content preference of their users. Fig. 9 depicts word clouds of visited domains (top row) and their corresponding tags (bottom row). We now identify those domains that are not commonly used. According to Fig. 9(a), we observe that the device 117 in Home-1 visits special contents such as gaming (e.g. `minecraft.net`) and educational (e.g. `stanford.edu`) domains thus making it more likely to be owned by a teenage member of the family. Similarly, we infer from Fig. 9(b) that device 478 in Home-2 is more likely to belong to an avid female social media user, since social media (e.g. `facebook.com`, `instagram.com`) and shopping (e.g. `taobao.com`) domains are visited frequently.



(a) Home 1. (b) Home 2.

Figure 8: Profile of data hungry devices.



(a) Home 1. (b) Home 2.

Figure 9: Profile of data hungry devices.

5 CONCLUSIONS

The explosion in Internet-connected consumer devices and the growing demand for Internet data is creating new pain-points for households grappling with an increasingly complex home network. We believe that this presents a business opportunity for ISPs to step-up and add value to the "dumb pipe" Internet service that they currently provide to subscribers. Our first contribution was to identify three such use-cases, related to quota management, parental controls, and smart-home security, that are in need of solutions in the near- to medium-term future. Our second contribution was to argue that SDN presents the right technology paradigm to deliver such value-add services at low-cost and large-scale. We presented an architecture in which the features are built and operated in the cloud, and the home gateway is relegated to an off-the-shelf device running open-source SDN firmware. Moving all the intelligence to the cloud allows features to be continually updated transparent to the home gateway and client devices, and allows customization of user-interfaces. Finally, we have implemented our platform as a commercial-grade product that is currently being piloted by an ISP in a developing country whose subscribers are lacking quota management tools. We have presented our analysis on how ISPs can benefit from visibility into household devices, their viewing patterns, and content preferences.

REFERENCES

[1] 2017. Asre Telecom. http://www.asretelecom.net/. (2017).
[2] Cisco Internet Business Solutions Group. 2012. Moving Toward Usage-Based Pricing. https://goo.gl/2YyUSD. (2012).
[3] Cisco VNI. 2017. Service Adoption Forecast for 2016 - 2021. https://goo.gl/SrpKbL. (2017).
[4] Daily Mail. 2016. Do YOU know what your child is up to online? https://goo.gl/nG1g3n. (2016).
[5] F. Loi et al. 2017. Systematically Evaluating Security and Privacy for Consumer IoT Devices. In *Proc. ACM CCS workshop on IoT S&P*. Dallas, Texas, USA.
[6] H. Habibi Gharakheili et al. 2016. Perspectives on Net Neutrality and Internet Fast-Lanes. *ACM Computer Communications Review* 46, 1 (Jan 2016), 64–69.
[7] M. Lyu et al. 2017. Quantifying the Reflective DDoS Attack Capability of Household IoT Devices. In *Proc. ACM WiSec*. Boston, Massachusetts.
[8] OpenDNS. 2017. Domain Tagging. https://community.opendns.com/domaintagging/. (2017).
[9] Statista. 2017. Average revenue per user of fixed broadband in Europe. https://www.statista.com/statistics/691685/fixed-broadband-arpu-in-europe/. (2017).
[10] J. Yang and W.K. Edwards. 2010. A Study on Network Management Tools of Householders. In *Proc. ACM SIGCOMM HomeNets*. New Delhi, India.