Smarter Home Networks with Simple Gateways

Sajad Shirali-Shahreza, Soheil Hassas Yeganeh, Yashar Ganjali
Department of Computer Science, University of Toronto, Toronto, Canada

***Abstract*—Users demand smarter home networks that conveniently provide networking features. Previous attempts focus on pushing more features into the gateways and making them smarter. Home gateways, however, have limited resources and cannot accommodate resource-demanding features. In this paper, in contrast, we delegate most gateway functionalities to a shared, consolidated control platform, and keep the gateway as simple as possible. Using this framework, we are not restricted by resource limitations of home gateways and thus are able to provide services such as device discovery or troubleshooting with less effort and better quality.**

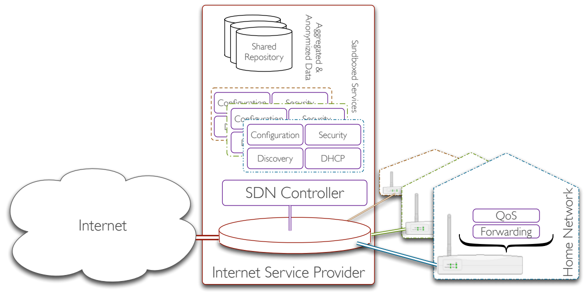

Fig. 2. Our Proposed Framework.

I. INTRODUCTION

There is a significant demand for smarter home networks with a pleasant user experience that not only provide basic connectivity, but also a set of features enabling users to make the best use of their network. Today's home gateways, however, barely afford basic networking functions, mainly due to the constraints in computing resources (i.e., small memories, tiny storage, and low processing power). Moreover, they have cumbersome development and maintenance processes that make it difficult to add novel features.

Previous works try to make the home gateway smarter (Fig. 1) to ease and enhance tasks that are currently possible (e.g., viewing current active devices [1]) and also offering new tasks and features that are not possible or available in current devices (e.g., bandwidth management [2]). The main challenge that they face is limited set of resources available on the home gateway: small memory, very small storage, limited processing power. As a result, they usually use a computer (e.g., laptop) to deploy and test their prototypes (e.g., [1] and [3]) and assume that future smart home gateways can implement them. Even the systems implemented on actual home gateways (e.g., [2] and [4]) mainly collect raw data on the device and send it to a server for processing (e.g., on a laptop inside home [2] or to a server via the Internet [4]) in order to overcome resource limitations.

Instead of trying to make gateways smarter, we propose to have a simple gateway coupled with a smart controller, and to move most functionalities from the gateway to the controller (Fig. 2). Software Defined Network (SDN) is a natural match to implement this idea. Utilizing SDN, we can delegate networking functionalities beyond forwarding to a centralized controller located in the closet vicinity of users (e.g., ISPs). In such a setting, home gateways become extremely simple accommodating an OpenFlow switch. At the same time, there will be more resources available in the controller, enabling us to provide services that are not feasible in previous settings. This will also ease future development.

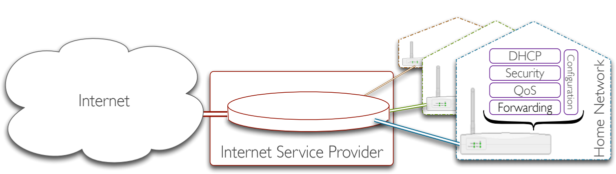

Fig. 1. Smart Gateway Architecture.

What differentiates our proposal from other SDN-based home networking solutions (such as [2] and [4]) is how functionalities are divided and where the controller resides. Prior works use the controller to complement smart gateways by delegating few tasks that cannot be done on the device. They either put the controller inside the home on a computer [2] or on a server that is not necessarily close [4]. In our proposal however, the controller resides on the service provider network, and is responsible for almost all network features.

In addition to well-known advantages of software defined networking (such as flexibility and programmability), this approach provides some unique opportunities for both users and service providers. For instance, it will significantly simplify the maintenance process as service providers will update most services in one place (not on every single connected home gateway), and will rarely push updates at the operating system level. As another example, sharing the same controller for all home users, operators can find more intelligent approaches for improving quality of service. Operators can automatically classify connected devices at the edge (home gateway) and enforce respective policies (e.g., high priority for VoIP packets) without the need for expensive throttling systems being deployed in the ISP.

Using our system, users can opt for networking features at will, with zero configuration and seamless updates; a luxury that is not straightforward to provide in today's home networks. Note that, such a remote, centralized control platform enables the operators to centrally maintain and update networking software that essentially reduce operating costs.

Concerns. The main concern about this approach is the hindered latency and resilience to failure. Considering the low latency between home networks and respective ISPs [4], remote services can serve the user within an acceptable latency range. Resilience can also be achieved through fallback mechanism in the gateway. When the controller is not reachable, the gateway can act as a learning layer-2 switch providing basic connectivity.

II. Simple Home Gateway, Smart Controller

Our ultimate goal is to make home networks smarter, but instead of making the gateway smarter by deploying new features inside it, we couple the home gateway with a smart controller in closest vicinity of home networks (e.g., ISP) and deploy these features on the controller. The gateway is essentially an embedded operating system equipped with an OpenFlow switch (Fig. 1). This box is responsible only for forwarding and QoS functionalities (NAT and BW management) readily available in OpenFlow. Services that need some programming logic, such as DHCP and device discovery, will be deployed on the shared control platform. We note that such home gateways, in contrast to existing ones, only need minimal configuration, i.e., connection info of the centralized controller. This makes it unlikely for them to be the source of any problem (e.g., misconfiguration or system halts), which could make troubleshooting profoundly easier.

We have implemented an early stage prototype of this system using NetGear WNDR-3700 access points running OpenFlow-enable OpenWRT, and utilized NOX as our centralized OpenFlow controller.

Isolation. Our control platform isolates network state of each user to preserve their privacy. Running services in user-specific sandboxes provides such isolation. In addition to the silos of isolated network state, services can share anonymized and aggregated data (such as classification rules and generic policies) in a centralized repository.

Legacy Services. To facilitate reuse, the proposed framework is designed with the capability to give services an illusion that they are running locally on one of the gateway. This approach is very similar to what has been proposed in RouteFlow. Using this feature, legacy services (e.g., existing DHCP servers) can be seamlessly deployed in our platform.

III. Sample Applications

We envision unique potential applications for our platform. In the interest of space, we present only two examples.

Device Discovery. Similar devices have similar communication characteristics and can work under generic policies. Thus, classifying devices is crucial for enforcing generic policies automatically in home networks. For instance, packets of a telephony device should be transmitted through high priority queues (automatic QoS policy generation), and a network printer only needs intra-network visibility for printing services (automatic access control instead of current unrestricted access inside home networks).

This important process sometimes needs complex deep packet inspection and machine learning mechanisms since some devices have no support for zero-config protocols (such as the UPnP protocol family) [5]. As a result, current home gateways either use very basic heuristics or do not provide discovery features at all.

Implementing device discovery in our platform is significantly easier. Services running on our platform are not limited in computing power, and thus can perform complex tasks. More importantly, we have the ability to observe numerous networking devices from multiple users and capture their characteristics. Such information can be utilized to perform much better device classification.

Troubleshooting. Troubleshooting, in today's home networks, is almost always reactive, is initiated by the end-user, requires a human intervention, is quite time consuming (e.g., 38 min. average time of a technical support call [6]), and usually fails to resolve the issue (e.g., user's problem is solved in only 14% of technical support calls, partially solved in 24% and not solved at all in 62% [6]). One of the reasons that remote technical support by technicians is hard and time consuming is because they cannot completely understand the problem and home network setup, mainly due to limited and inaccurate information that home user provides [6].

In a smarter environment, the system can automatically collect and process required information to identify and solve problems with minimal technician or home user interaction. But, such a smart service needs to store detailed networking events and analyze them to discover patterns. Today's home gateways do not have enough processing and storage resources for such an intensive task. More importantly, without having the history of large number of users, it is difficult to detect failure patterns.

As our platform has profoundly more persistent storage space than a home gateway, services can easily store detailed events for an arbitrarily long duration in an isolated, privacy preserving storage. Service providers can utilize network logs in order to proactively detect flaws in home networks and resolve them before users notice and react.

IV. Final Remarks

Eliminating resource constraints, the proposed framework paves the way for futuristic networking services that are out of the picture in today's home networks. We envision this framework to be the groundwork for numerous automatic optimizations. Having that in mind, we are extending our prototype to offer most basic networking services with ultimate automation at scale. Moreover, we are redesigning user interfaces with the help of the data and processing power offered by the control platform.

References

[1] J. Yang, et al., Eden: supporting home network management through interactive visual tools. UIST 2010. 109-118.

[2] M. Chetty, et al. Why is my internet slow?: making network speeds visible. In CHI 2011, 1889-1898.

[3] K. L. Calvert, et al., Instrumenting home networks. SIGCOMM CCR, 41, 1 (January 2011), 84-89.

[4] S. Sundaresan et al. Broadband internet performance: a view from the gateway. In SIGCOMM 2011, 134-145.

[5] L. DiCioccio et al., Characterizing home networks with HomeNet Profiler, Technicolor, CR-PRL-2011-09-0001, 2011.

[6] E.S. Poole, et. al, The Home Network as a Socio-Technical System: Understanding the Challenges of Remote Home Network Help, JCSCW, 18, 2-3, 2009