Characterising Usage Patterns and Privacy Risks of a Home Security Camera Service

Jinyang Li, Zhenyu Li, Gareth Tyson, and Gaogang Xie

Abstract—Home security cameras (HSCs) are becoming increasingly important in protecting people's household property and caring for family members. As an emerging type of home IoT devices, HSCs are distinct from traditional IoT devices in that they are often installed in intimate places, detecting movements constantly. Such close integration with users' daily life may result in distinct user behavioral patterns and privacy concerns. To explore this, we perform a detailed measurement study based on a large-scale service log dataset from a major HSC service provider. Our analysis reveals unique usage patterns of HSCs, including significant wasted uploads, asymmetrical upload and download traffic, skewed user engagement, and limited watching locations. We further identify three types of privacy risks in current HSC services using both passive logs and active measurements. These risks can be exploited by attackers, through observing only the traffic rates of HSCs, to infer the working state of cameras and even the daily activity routine in places where the camera is installed. Moreover, we find the premium users who pay an extra fee are especially vulnerable to such privacy inferences. We propose countermeasures from the perspectives of susceptible users and HSC providers to mitigate the risks.

Index Terms—Home security camera; IoT; Privacy; Usage pattern.

1 INTRODUCTION

The Home Security Camera (**HSC**) is a home wireless Internet of Things (IoT) device that has become increasingly common in people's daily lives in recent years. With the entry of major players such as Nest [8], XiaoMi [11], Hikvision [5], and Netgear [9], the HSC market is expected to reach $1.3 billion by 2023 [6]. The home security cameras are often installed in private places (like home or office) for securing the safety of the property or caring for the elders and kids. The cameras are connected to the Internet via WiFi. The camera owner can watch the live feed through the mobile app provided by the camera provider from everywhere. The video content is transmitted via the cloud servers of the provider — the content is uploaded to the servers first and then stream to the mobile app. Besides the live streaming mode, HSCs often provides a motion detection mode, where the HSC will alert the owner through the mobile app when movements being detected in front of the camera. Most HSC services provide the third mode, which upgrades the second mode, called *Replay mode*, a functionality exclusively available for premium users (who pay an extra fee). With the replay mode, the HSC will record a video clip for the movements, and upload it to the cloud. The owner (premium user) can later replay the clip to examine the movements.

As a new type of Internet video, the HSC service providers and the network operators are eager to know the usage patterns of HSCs, in order to understand the impact on network load and optimise the systems. While there are a load of studies on measuring the usage behavior of other Internet video systems [21] [22] [28] [25] [33], their findings may not be applicable to HSCs. This is because of the unique features of HSCs, such as the replay mode, usage context, and unicast communication between cameras and owners.

Another important aspect relevant to HSCs is about the privacy risks (if there is any) as they are often installed at intimate places. HSCs, once being turned on, will monitor every movement that occurs in the installation places 7×24, and will automatically upload recordings when movement is detected (a notification for normal users and a clip about the movement for premium users). Despite that the content is encrypted, it may be possible for attackers to infer private information from the traffic patterns (*e.g.* traffic rate) as previous studies have proved this possibility for other types of IoT devices [1], [15], [18], [40]. For most people, it is difficult to verify if protections are in place [49], and thus, this offers strong motivation to investigate any potential leakage of user privacy brought about by the HSC. Despite some initial studies on the possible privacy leakage of HSCs or alike services [18], [31], we still lack a comprehensive understanding of the privacy risks of HSCs using both large-scale passive logs and active measurements.

Thus, we argue that the novel features of the HSC warrant further investigation. We are particularly interested in the following questions: (*i*) What are the user behavioral patterns of HSC's live streaming mode and replay mode, respectively? (*ii*) What are the differences between the usage patterns of premium and normal users? (*iii*) Are there any privacy risks, and could a tractable adversary exploit them?

- *J. Li and Z. Li are with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China, and University of Chinese Academy of Sciences, Beijing 100049, China. Z. Li is also with the Peng Cheng Laboratory, Shenzhen 518000, China. E-mail: {lijinyang, zyli}@ict.ac.cn*
- *G. Tyson is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, United Kingdom. E-mail: g.tyson@qmul.ac.uk*
- *G. Xie is with the Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China. E-mail: xie@cnic.cn*

Corresponding author: Zhenyu Li.

(*iv*) What mitigation would address these privacy concerns?

To answer these questions, we rely on a unique dataset of passive service logs collected from a mainstream HSC provider for a week period. The dataset covers 15.4M streams from 211K active users (§2). It contains a mix of premium and normal (free) users, allowing us to explore a wide diversity of HSC behaviors. We complement the passive logs with active measurements of three popular HSCs, in order to have a deep understanding of the working flows and generalize our findings as well. Our analysis starts by examining the overall behavioral patterns of the examined HSC service. We then proceed to perform unsupervised clustering to identify key user types, followed by diving into the watching locality (§3). We then explore a set of privacy attacks and characterise their efficacy (§4), discovering a subset of highly regular users for whom the attacker can effectively infer their activities. Our key findings include:

1) The platform is dominated by premium users, who also produce a large volume of wasted content. Premium users constitute 59% of all accounts, yet contribute more than 95% of total traffic, since they are much more active in using live streaming than normal users and the replay mode is exclusive for them. Despite this, 60% of the total video data is wasted, since a great amount of motion-triggered replay goes unwatched. This waste is largely attributed to a handful of very heavy premium users ($\sim$1/4).
2) The viewings locations are predictable: about 10% of users appear to utilize the HSCs as a regular surveillance service and generate a huge amount of viewing traffic. Such users tend to view their HSC streams from 1 or 2 key (network) locations, and often these are at a different location to the camera. This is likely driven by the differing uses of HSCs, ranging from monitoring children' s safety to surveillance of commercial properties.
3) We identify three major privacy risks: (*i*) traffic surge risk, (*ii*) traffic regularity risk, and (*iii*) traffic rate change risk. These attacks allow an adversary to predict the daily patterns of the camera uploads and even infer activities on the camera feeds. We propose methodologies to infer privacy-compromising information, and explore the risks with both the passive logs and active measurements of three popular HSCs.
4) We find that, premium users are more vulnerable to privacy risks because of their heavier live video usage and exclusive access to the additional replay mode. For example, the accuracy of predicting the patterns of premium users' upload streams is as high as 0.75 (3$\times$ the accuracy for live streams by non-premium users). Moreover, we propose to obfuscate video uploads to minimize the exposure to these attacks.

While the basic user behavioral patterns and privacy issues have been presented in [32], the workload diurnal patterns and user engagements (relevant to user behavior patterns), as well as the user activity switch patterns (relevant to privacy risks) are newly added in this extended version.

The remainder of this paper is organized as follows. Section 2 describes the background and dataset. Section 3 presents an in-depth analysis of user behavior, while Section 4 investigates the privacy leakage problem. We discuss the related work in Section 5. Finally, Section 6 concludes our work.

2 Background & Dataset

In this section, we first introduce the HSC service, then detail the dataset we utilize.

2.1 Primer on Home Security Cameras

We first briefly explain the operating procedures of typical HSC services. Upon purchase, the owner of an HSC first downloads the corresponding HSC app on a smartphone and then connects the HSC via an accessible Wi-Fi. The user then binds the camera to her account. The camera passively receives commands from the servers that are hosted in a cloud operated by the HSC provider. After setup, the user can remotely request a live stream or an archived replay via the cloud servers using the mobile app. It is worth noting that users *never* connect directly to the camera — all video traffic is forwarded via the servers.

HSC cameras of major HSC providers (*e.g.* Nest, Netgear, Hikvision, and 360) often support two modes of streaming:

- *Live streaming mode*: The user is able to login and initiate a live stream from the camera in realtime, via the cloud server as an intermediary. The video will *not* be stored anywhere by default.
- *Motion detection mode*: When a motion is detected, an app notification is sent, and the user is then given the option of viewing the stream in real time. Again, nothing will be stored by default.

As the above modes are inconvenient for users who cannot immediately view streams in realtime, some HSC services offer another feature for premium users (who pay a fee), where the motion detection mode automatically uploads and stores motion-triggered streams to the cloud servers. These streams contain video footage from a few seconds before the motion begins, until a few seconds after. Premium users can then replay the video at any time, and a video will be saved for several days. We term this *replay mode*.

2.2 Dataset Description

Our work relies on two datasets: (*i*) a large scale service log dataset from a major HSC service provider, and (*ii*) actively collected network traffic data of 3 mainstream HSCs.

Passive service log: Our service log dataset is a 7-day dataset of log entries shared by a major Chinese HSC service. The examined HSC provider serves hundreds of thousands of users per day and supports all the above features. The dataset covers *all* cameras that were connected to the Internet via a major ISP in China. Every individual log related to these cameras is included in our dataset.[1]

Within the logs, one video view or upload corresponds to one *stream*. A service log is generated for every 30-second

1. This includes cases where a user is viewing the camera feed from a different ISP.

segment for each stream, so it is reasonable for 1 stream to be related to more than 1 log. In total, we obtain 96,515,229 logs of 15,432,950 streams from 211,806 unique active users that have uploaded at least once (either live stream or replay videos) during the observation period. Of these users, 124,985 (124K) are premium (who pay a fee), accounting for 59% of all active users. The remaining non-premium users are referred to as *normal users* throughout this paper. Note that while both types of users have unlimited access to live streaming, replay mode is only available for premium users.

Each log entry, which corresponds to a 30-second segment, includes three main categories of information:

1) *User-specific information*: the anonymized user ID that is uniquely bound to a registered account, as well as this user's camera(s); the IP address (anonymized using `Crypto-PAn` [2]); the anonymized BGP prefix of the IP address, which is obtained by querying `Team Cymru` [10].
2) *Stream-specific information*: the anonymized stream ID; stream type (*up* for video uploading, *down* for video viewing).
3) *Segment-specific information*: the average bit rate of this segment (kbps); data volume (KB); and timestamps that mark the start and end of the segment.

Active measurement traces: We note that the above data pertains to a particular HSC provider. To this end, we set up a testbed to capture the packet-level traces (128-byte packets) actively of three popular HSC services: Nest HSC (popular in Western countries), XiaoMi HSC (popular in China), and also the examined one in this paper. The cameras are connected to the Internet via a laptop that acts as a Wi-Fi access point. We use Wireshark to capture all packets relevant to each HSC under both live stream mode and motion detection mode. The traces enable us to have a better understanding of the privacy risks in HSC services.

Ethical Issues: We took a number of steps to ensure the ethical use of the data shared with us. We have no access to the content of video streams, and can only observe metadata (*e.g.* stream duration). The logs used are routinely gathered for operational purposes, and no extra data collection was triggered. All user information, including user ID, IP address, BGP prefix, and even the stream ID, is fully anonymized. We are unable, and not allowed, to link logs to users. We also leverage volunteers for controlled experiments (§4.4), where the cameras were placed in working areas (rather than homes). The volunteers were aware that we only use traffic rate information. Finally, we have reported all potential privacy risks to the service provider and assisted them in implementing fixes.

3 EXPLORING USER BEHAVIOR

Before investigating privacy issues, it is first necessary to understand user behavior. Here, we present a characterization of typical usage patterns in the examined HSC service.

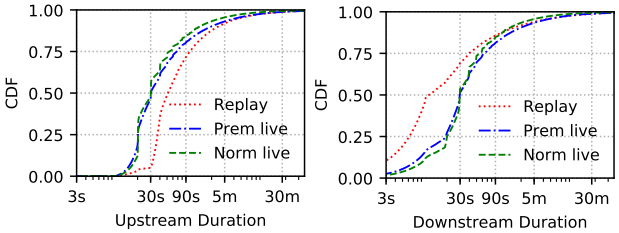
3.1 Basic Characterisation

Stream Volumes:

We first inspect the data volumes uploaded/downloaded by each user and traffic type.

TABLE 1: Data volume distribution.

	Normal user	Premium user		
	live stream	live stream	replay	All
Up	1.37%	12.96%	65.89%	80.22%
Down	1.36%	12.77%	5.65%	19.78%
All	2.73%	25.73%	71.54%	100%



(a) Upstream (b) Downstream

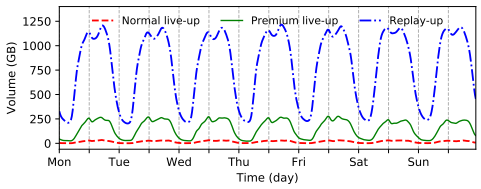
Fig. 1: Distribution of stream duration.

Note that all upload (*up*) streams are initiated by a camera uploading data to the server, whereas all download (*down*) streams are initiated by a user viewing the video. Live-up and live-down are generated in pairs when users request live feeds from the camera. Recall that the *replay* mode is only available to premium users: the replay-up streams are exclusively triggered by motion seen by the camera, while the replay-down streams are triggered by premium users watching the replay videos, which may happen at different times than the replay uploading.

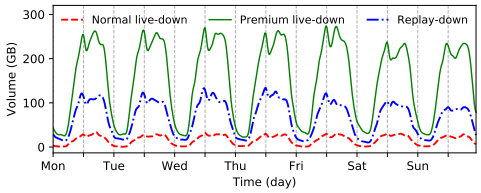
Table 1 summarizes the results. As expected, live-up traffic matches the live-down traffic. This is because a live-up stream exactly corresponds to a live-down stream. We can also observe that the platform is dominated by traffic generated by services for supporting premium users. The premium accounts tend to be heavy users: they generate 97% of the traffic. This is caused by heavier use of live streaming by premium users, and more importantly, by the dominance of motion-triggered automatic uploads — replay-up streams contribute over 2/3 of the total workload. As a striking contrast, only ∼5% of download streams come from this source (replay). In fact, we see that a remarkable 60.24% of video uploaded is never downloaded, suggesting a significant waste in both network and storage resources. This is particularly because replay-up streams, on average, last longer and have larger volumes (median around 4MB). In contrast, the replay-down streams are shorter and with smaller sizes (0.65MB). This drives the asymmetry of the workload in Table 1. We will examine the wasted traffic in detail later.

Stream Duration: Figure 1 illustrates the distribution of stream duration for both upload streams and download streams. We can see that whereas HSCs tend to upload the long motion-triggered video (median 50 seconds), users tend to only replay them for a short duration (median 10 seconds). We conjecture that such users are only checking to ascertain the reason for the motion — once this is established the viewing is cancelled. The above asymmetry in stream volume and duration is one of the reasons that contribute to the asymmetry of workload in Table 1.

Diurnal Patterns: Figure 2 further presents the variation

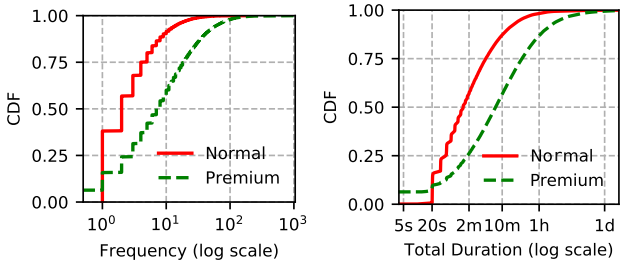


(a) Upstream



(b) Downstream

Fig. 2: Temporal variation of the workload of the examined HSC service. The vertical dashed represents 0:00 and 12:00 of the day.



(a) Frequency (b) Duration

Fig. 3: Distribution of active users' live streaming usages.

TABLE 2: User clustering for live streaming.

	no.	freq.	dura.(s)	%	feature
Normal	#1	1.5	41.7	49	Light
	#2	4.8	287.1	42	Medium
	#3	16.9	3,719.1	9	Heavy
Premium	#1	3.4	129.5	45	Light
	#2	18.4	1,422.8	44	Medium
	#3	64.6	22,217.2	11	Heavy

of traffic volumes over one week, where again we observe the dominance of premium users. The figures are separated into different stream categories, although the trends across different categories of streams are broadly similar. Each shows clear diurnal patterns, where more data is transmitted during the daytime with 2 clear peaks around 12:00 and 17:00, implying more intensive motions at those time points. The workload pattern we observe in HSC is therefore different from that of mobile VoD service [34], which often hits a major peak around 23:00. Additionally, it is also not the same as live streaming service [44] or personal streaming [39], which reaches their peak in the late afternoon. Therefore, workload optimization schemes designed for traditional VoD or live streaming services may need to be re-examined before they can be applied to the HSC service.

3.2 Characterising Live Stream Mode

Next, we inspect the generation and consumption of live content by users. Note that these include *both* normal users and those with premium accounts.

Overview of Live Users: We first count the number (termed *frequency* hereafter) and the *total* duration of the live-down streams generated per user. We show the cumulative distribution in Figure 3.[2] As expected, premium users are more active than normal ones: the median frequency of streams is 7 for premium users *vs.* 2 for normal users. This observation is also mirrored when inspecting duration: the median total duration of normal users and premium users are 90 seconds and 435 seconds respectively. Nevertheless, some normal users generate over 100 streams and watch over 5 hours during the observation period.

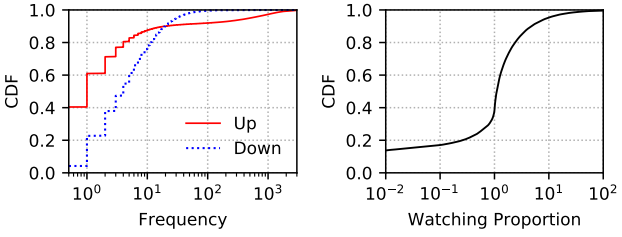
Clustering Live Stream Users: The above suggests a diversity of user groups. Thus, we proceed to identify core behavioral types within the user population. To this end, we fit the *frequency* and *total duration* statistics of all users to a 3-component Gaussian Mixed Model (GMM) [4].

We experimented with a number of configurations from 2 to 5 GMM components and selected 3 based on the balance between relatively small AIC (Akaike Information Criterion) and not too small components ($percentage < 0.1\%$). Table 2 presents the clusters identified, alongside their fitting results. The results expose three main sub-populations, shared across both normal and premium users. We term these *light* (L), *medium* (M), and *heavy* (H). Light users use the live streaming service rarely. In contrast, medium users tend to check their camera feeds daily. The heavy users deviate significantly from the average, with extremely regular viewing patterns. It seems likely that heavy users use HSCs as a (potentially commercial) surveillance camera service, and the cameras likely cover high-value regions (*e.g.* in a shop).

In summary, while the live streaming mode is presented to both premium and normal users, premium users use it more heavily than normal users.

3.3 Characterising Replay Mode

We next inspect users of the video *replay* service. This is *only* available to premium users (59% of the population). When activated, the replay mode automatically uploads all motion-triggered content to the cloud for later on-demand access.



(a) Frequency (b) Watching Proportion

Fig. 4: Per-user characteristics of replay mode.

2. Note that upload and download streams are approximately symmetrical in terms of frequency and duration in the case of live streaming.

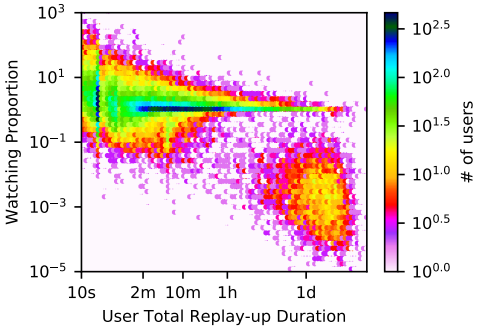


Fig. 5: Distribution of premium users, based on their watching proportion and total replay-up duration. The scales of axes and color bar are logarithmic.

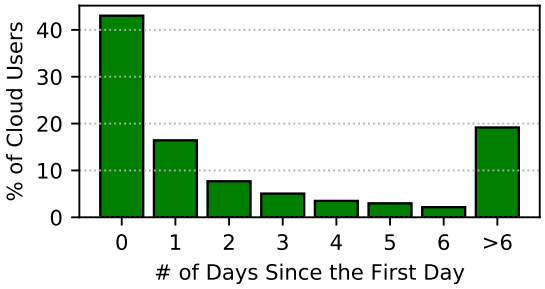


Fig. 6: Fraction of users that upload replay videos on the first observation day and watch them on the x-th day (relative to the first day).

Overview of Replay Users: Figure 4(a) presents the number of replay streams per user. It shows that viewing replay content is significantly more frequent (per-user) than uploading. This was surprising as Table 1 found that the majority of data is generated by replay uploads. A deep investigation reveals that whereas, by volume, the majority of data uploaded is replay traffic, this is driven by a small subset of streams. A remarkable 40% of (premium) users *never* upload a video for replay during the week, leaving a small subset of users with exceedingly high upload rates — 97% of the total replay upload traffic is generated by the top 5% of premium users. This indicates a mix of camera installations, with many fitted in locations with very limited motion and others in high activity zones.

This disparity raises the question of whether users actually view the videos uploaded. To measure this, we define the *Watching Proportion* as the ratio of a user's total replay-down duration to total replay-up duration.[3] Figure 4(b) depicts the distribution of watching proportions. While the median is around 1, it is less than 10^{-2} for 13.8% of users, meaning that they watch far less than their cameras upload. Nevertheless, about 61.5% of premium users have a watching proportion in excess of 1, indicating that they repeatedly watch the same streams.

Figure 5 further examines the correlation between the uploaded volume and the watching proportion. We can observe two typical patterns. Most users are within the first group (upper left), where the total uploaded duration is relatively low, yet the watching proportion is relatively high (around 1). The remaining users (about 1/4 of the total premium users) are within the second group (lower right), where the total replay-up duration is relatively high, but the watching proportion is relatively low. The actions of these users result in over 60% of network and storage resources being wasted.

In summary, the huge traffic waste relevant to the replay mode is largely attributed to a handful of very heavy premium users (∼1/4) that reside in the lower right in Figure 5. Indeed, only 4% of premium users never watch replay (See Figure 4(a)), as many as 61.5% of the premium users watch at least the same length of the uploaded replay streams (see Figure 4(b)). Our results suggest that the service providers should adopt a more informed upload strategy to meet user needs while saving costs.

User Engagement: We next inspect user engagement, in terms of how long it takes a user to view a video once it has been uploaded. One would imagine that HSCs streaming (potentially) important content would warrant immediate viewing. Figure 6 presents a histogram of the number of days it takes a (premium) user to view an uploaded replay video. Over 40% of premium users check the video feed the same day as the upload. This is intuitive in cases where the cameras are being used for security purposes. Nevertheless, as many as 20% of users never come back within the observed week. This suggests that many users do not actually benefit from the automated motion-sensitive uploads.

Clustering Replay Users: The above shows a wide range of behavioral types. Thus, we repeat our earlier user clustering process. Here, we use total replay-up duration and watching proportion of active premium users to fit a 3-component GMM.[4] The fitting result is shown in Table 3. This exposes three broad categories of users, which we index as *light* (L), *medium* (M), and *heavy* (H) for watchers (W) or uploaders (U), respectively. Thus, each user cluster is tagged with both the watching and upload behaviours.

TABLE 3: Clustering active premium users.

no.	up dura.(s)	watch. prop.	%	feature
#1	46.8	20.6	11	LU-HW
#2	298.7	1.4	65	MU-MW
#3	94,823.3	0.5	24	HU-LW

The majority (around 2/3) of premium users fall into the Medium Uploader and Medium Watcher category (MU-MW). They keep the best balance between upload and watching rates. The next most populated group are those who are Heavy Uploaders but Light Watchers (HU-LW). Such users are most costly to the system, as they consume large amounts of network and storage, yet do not benefit from them. At the opposite extreme, the smallest group are those that have a low rate of uploads, but a high rate of watching (LU-HW).

3.4 Characterising Viewing Locality

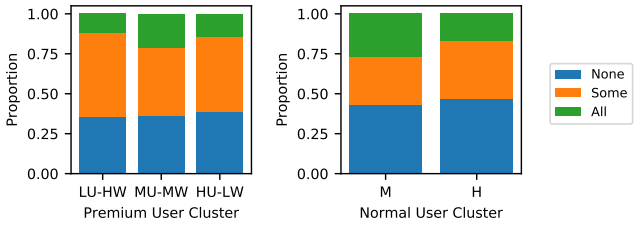
We next proceed to explore *where* streams are uploaded and consumed from. This is particularly important for QoE

3. We only include premium users who have uploaded replay video for at least 10 seconds.

4. We decided on the component number here using the same approach as mentioned in §3.2.

improvement, *e.g.* via caching or pre-fetching.

On-site* vs. *Off-site Access: Since we are interested in the network footprint of users, we use the BGP prefix of the user's IP address to represent the user's location. We represent proximity as a binary metric where we test if the camera and viewer are located within the same prefix. We refer to accesses that occur from the same prefix as the camera as *on-site*, and similarly, we denote accesses that occur from a different prefix as the camera as *off-site*.



(a) Premium user (b) Normal users

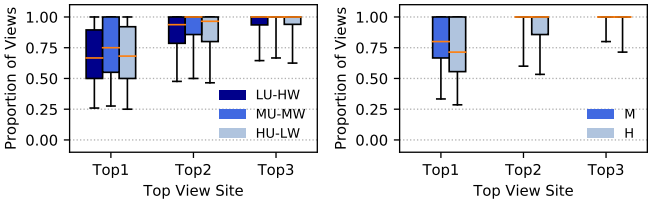
Fig. 7: The proportion of users who have watched a video from the same BGP prefix as the camera for none, all or some of the views in each user component.

To inspect the locality patterns of the different user groups, we take the premium user categories from Table 3, as well as the normal user categories from Table 2. We exclude any users who have fewer than 3 views in the observation period, to avoid bias caused by the sparse sampling of irregular users. Figure 7 presents the breakdown of on-site *vs.* off-site views for each group of users. *None* indicates that no view comes from the same prefix; *some* indicates that a fraction (> 0) of views come from the same prefix; and *all* indicates that all user views emanate from the same prefix as the camera.

Users exhibit similar behaviors across all usage groups. About 30% of examined users consume no streams on-site. This is rational, as there is perhaps little sense in accessing camera feeds from the same site in many cases. The remaining users may experience local access under several situations where a single site covers a large area (*e.g.* factory) or where users employ cameras for monitoring local activities (*e.g.* sleeping children).

User Mobility: We finally inspect how mobile users are, *i.e.*, whether users always view from the same location. To this end, we compute for each user the proportion of views that happened at the top k locations, where $k \in \{1, 2, 3\}$. Users are again grouped based on the earlier clustering results.

Figure 8 presents the fraction of views from the top k locations per-user as a box plot. The majority of users view primarily from their top 1 or 2 locations. The median fraction of views in the top 1 location and top 2 locations for premium users is about 0.7 and 0.95, respectively. Although normal users tend not to watch the live streams on-site (see Figure 7), they are more likely to view the content at a single location than premium users: over 40% of normal users watch all streams at the top locations, while this number for premium users is only about 30%. This indicates that users may not move often (*e.g.* staying at their office). These observations imply the possibility of predicting users' next viewing location and perform pre-loading of material for improved QoE.



(a) Premium user (b) Normal user

Fig. 8: The proportion of views that happened at the top k locations for each user, where $k \in \{1, 2, 3\}$.

3.5 Takehome Messages

We make three notable observations: (*i*) *Wasted resources*: premium users generate the majority of the workload (97.27%), largely due to the exclusive availability of replay mode. This results in 60% of the uploaded videos going unwatched. The waste is attributed to a handful ($\sim$1/4) of heavy premium users, who have cameras with high levels of motion-triggered uploads (see Table 3). (*ii*) *Distinct usage patterns*: premium users show heavier live streaming usage pattern and higher levels of engagement than normal ones; these account types have a mix of access patterns. (*iii*) *Watching locality*: users tend to view streams from 1 or 2 key locations, with a sizeable portion of users watching streams from individual remote sites, suggesting a surveillance use case.

The above indicates that a set of simple innovations could streamline HSC operations. Most notably, HSC services could benefit from on-demand (rather than real time) uploads for the replay mode. This is because most unwatched replays come from a handful of heavy premium users and last for a longer time. Therefore local recording (with on-demand uploads) could offload a significant volume of unwatched uploads from the network. There are a number of ways this could be implemented without negatively impacting the customer experience. For example, HSCs could upload the first 1 minute of a stream by default, allowing users to 'preview' the content. If a user wishes to continue viewing (this applies to less than 20% of streams), the remaining video can be requested from the HSC. This could also be mixed with more sophisticated methods of delivery, whereby only users predicted to consume content have their previews uploaded. The predictability of *where* users view content from means these videos could even be pre-fetched (*e.g.* to the top 1 or 2 locations). This will reduce the startup delay, and thus improve the QoE.

4 Exploring Privacy Risks

HSCs are always-on sensors devised to actively detect movements mostly in intimate places. The motion-triggered recordings are automatically uploaded. As such, network traffic patterns may expose information about users. This differs substantially from other traffic-inference attacks (*e.g.* monitoring a user's Netflix usage) as variations in the camera feed (*e.g.* bit rate changes) can expose specifics

about behavior, such as exposing Activities of Daily Living (ADL). In this section, we study potential privacy risks and solutions.

4.1 Adversary Model

Our adversary is able to monitor all network traffic in and out of home gateway routers via Wi-Fi sniffing (*e.g.* `DeWiCam` [23]), and then utilize IoT stream classification approach to separate the HSC streams from others, which is well studied in [14], [18], [27], [38], [41]. In most cases, the payload is encrypted; nevertheless, the attacker can exploit and *only* needs metadata, *i.e.,* IP packet headers and traffic rate. We also assume a targeted attacker, who has an approximate understanding of the camera's context (*e.g.* if it is mounted in a house), and who owns it. Based on our adversary model, we identify three major privacy risks:

1) *Traffic surge risk*: If the traffic rate of a camera surges from its base rate, this indicates that the video is being uploaded. In the case of motion detection mode, this indicates activity near the camera zone.
2) *Traffic regularity risk*: After a period of observing surges, an attacker may be able to infer a user's daily patterns. For example, a camera consistently uploading motion-triggered video at 18:00 might indicate that family members arrive home at that time.
3) *Rate change risk*: The different activity patterns of the photographed subject will result in different HSC traffic rates. This is because variable-bitrate (VBR) encoding is often used for video compression, where the bit rate of a video stream is closely related to the video content. Based on these rate variations, an attacker may be able to infer the intensity of activity being undertaken, and even the types of activity.

The first risk and the third risk have been examined in [18], [31] via small-scale testbeds using active measurements. We extend the analysis using our passive logs and active measurements of three popular HSCs. The second risk is explored for the first time. It should be noted that although the examined HSC is based in China, these attacks are equally applicable to other cameras that use VBR encoding, including Nest [18] as we show later in this section.

4.2 Traffic Surge Risk

A *traffic surge* is a point in time where the bit rate of a camera's feed increases dramatically. This creates a privacy risk that is inherently part of the transmission schemes of HSC services (since the camera only uploads when some certain functions are activated and stays idle for the rest of the time). This constitutes the foundation stone of all subsequent attacks.

Methodology: A traffic surge may be triggered by one of two events: (*i*) a user viewing the live stream; or (*ii*) the motion capture triggering an upload for later replay (in the case of premium users). In both cases, once an upload is triggered, a significant surge in traffic is observed. Trivial peak identification across the bit rate time series can therefore be used to verify if the camera is recording.

It is noteworthy that motion-triggered uploads correspond to the real time movements in private places, while live uploads may be triggered by the viewing requests of users. As such, an attacker who can differentiate between live and motion-triggered uploads is threatening. Inspection of the traffic reveals that it is possible to differentiate between live and motion-triggered uploads. This is because motion-triggered uploads always start with an initial peak, due to the uploading of a motionless video at the beginning of the transmission. We can identify peaks from the traffic rate time series, $S = \{s_1, s_2, ..., s_n\}$, where s_i represents the bit rate observed at the time point i. The maximum value within S is denoted as p. We assume the traffic rates follow a Gaussian distribution, then we construct a second set with p removed, $S' = \{s | s \in S, s \neq p\}$. We calculate the standard deviation (σ) and the average of S' (*i.e.,* $\bar{S'}$). Values less than $1.96\sigma + \bar{S'}$ occupy approximately 95% of the probability space [13]. If the maximum value within S is greater than $1.96\sigma + \bar{S'}$, this time series indicates a replay upload. Otherwise, it indicates a live stream. We have also verified the presence of this traffic surge risk in two other HSCs: XiaoMi and Nest.

Risk Exploration: To measure the efficacy of this risk, we conduct a set of controlled experiments, where we connect three types of HSCs (namely an examined HSC, a XiaoMi HSC, and a Nest HSC) using a laptop as the Wi-Fi access point, and perform packet capture using Wireshark. We leave the cameras dormant before starting to view the stream after 25 seconds. Figure 9(a), 9(c) and 9(e) present the time series of the normalized bit rate for three types of HSCs respectively. When we start to watch a live stream, all cameras switch from motion detection mode (white) to live streaming mode (green): this is shown by the sudden spike in bit rate. When we finish watching, all cameras switch back to motion detection mode and stop the transmission.

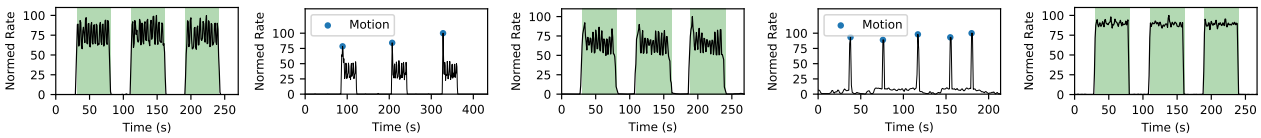
To confirm our ability to differentiate live and motion-triggered uploads (*i.e.,* replay-up), we repeat the above setup[5] with premium user accounts and periodically simulate motion in front of the cameras. Figure 9(b) presents the result for the examined HSC, confirming that motion triggers an immediate upload. For the Nest HSC in Figure 9(d), we can also observe clear traffic spikes when the motions were triggered. Importantly, the traffic peaks at the beginning of transmissions are notable and trivial to detect using our methodology (*i.e.,* we obtain 100% accuracy).

4.3 Traffic Regularity Risk

The above shows that a passive attacker can infer (*i*) if a camera is uploading; and (*ii*) if that content is being streamed for motion capture replay. We next explore if this can be exploited to identify regular patterns in a user's behavior. For example, if a camera in a house regularly initiates a motion-triggered stream at 7 AM, an attacker could infer that this is the time the owner awakes. Such information could be used to enable physical attacks, *e.g.* burglary.

Methodology: To test if such regularity can be inferred from network traffic, for each user, we define the *Regularity Value* (RV) as follows. We first count the upload duration per

5. We exclude XiaoMi HSC, since it did not support motion detection mode when the experiments were conducted.



(a) Examined HSC live-up (b) Examined HSC replay-up (c) Nest live-up (d) Nest motion detection (e) XiaoMi live-up

Fig. 9: Normalized traffic rates for the examined HSC, Nest HSC and XiaoMi HSC in live streaming mode and motion detection mode.

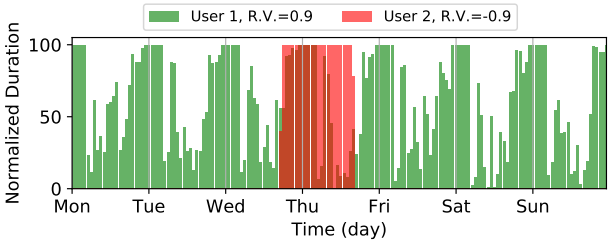
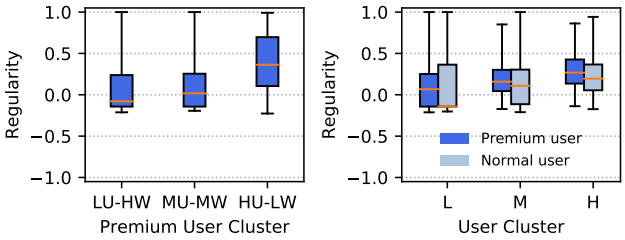


Fig. 10: Durations of premium users uploaded replay video and corresponding regularity value (R.V.) example.



(a) Replay video (b) Live streaming

Fig. 11: Distr. of regularity value of different clusters of users.

hour across the observed period. This yields one 24-element vector per day. We then filter out the days without any uploading. For each of the remaining vectors, we compute the moving average with a window size of 3 hours, in order to compensate for variations in a user's daily activity (*e.g.* viewing a stream at 10:03 rather than 9:59). We then calculate the pairwise Pearson Correlation Coefficient between all possible pairs of a user's daily vectors; we refer to the final per-user average as the regularity value.

The regularity value ranges from -1 to 1. If the value is positive, the upload patterns of the user are regular (the closer to 1, the stronger the regularity). This indicates the daily patterns of the user can be inferred. Note that the regularity value is not an attack in itself — it quantifies how susceptible users in our dataset are to this type of analysis by an attacker. To highlight the efficacy of regularity value, Figure 10 presents the hourly duration recorded across two users in our dataset. The duration being captured are for the motion-triggered uploads. User 1 (green) exhibits highly regular patterns, with a score of 0.9. It can be seen that this user's camera performs motion-triggered uploads regularly at night and during the early morning. In contrast, User 2 (red) has highly irregular patterns (score of -0.9), where all camera activity is across several hours on Wednesday and Thursday.

Risk Exploration: We next test the regularity of users in our dataset. We only inspect those users who have uploaded for at least 2 days within the observed week. This includes 67% of all users who have performed live uploads, and 51% of premium users who have motion-triggered replay uploads. Figure 11 presents the distribution of regularity values across normal and premium users. Again, we separate these users into their categories as identified via our earlier GMM clustering.

Figure 11(a) reveals significant diversity across the different clusters for motion-triggered replay predictability. The majority of LU-HW users and MU-MW users show little-to-no regularity. Their median regularity is near to 0, indicating that their daily patterns are difficult to predict. This is partly driven by their very nature, which consists of limited usage. In stark contrast, HU-LW users show stronger regularity, with the 75-th percentile as high as 0.69. In the case of live videos (Figure 11(b)), in all clusters, the regularity of premium users is higher than that of normal users. Furthermore, as the usage frequency becomes higher (for both premium and normal users), the regularity becomes higher. This intuitive finding indicates that heavier users are easier to predict.

To find a reasonable threshold ($thresh$) for what might be considered *strongly predictable*, we use the approach suggested in [29]. We fit all positive user regularity values into a 2-component GMM, and we define the intersection of the 2 resulting components as the threshold. The resulting value of $thresh$ is 0.34 in both the cases of normal and premium live uploading, and 0.35 in the case of replay uploading. Consequently, 17.4% of replay-up premium users, 18.7% of live-up premium users and 11.5% live-up normal users can be considered as strongly predictable.

Predicting Activity: We next confirm that this regularity can be exploited to predict upload patterns. We extract all users who have uploaded video data on *all* 7 days: 9,912 (8%) premium users for replay-up and 14,826 (7%) users for live-up. We then use their first four days to fit a Holt-Winters model [3] to predict the following three days time series (seasonal period was set to 24). Note that the time series are binary (0 for no upload in that hour, 1 otherwise). We compute the prediction accuracy as follow:

$$Hit\ rate = 1/n * \sum_{i=1}^{n} hr_i \quad (1)$$

where hr_i is 1 if the prediction for the i-th time slot is correct (0 otherwise), and n is the number of hours under prediction (*i.e.*, $3 * 24$).

Figure 12 presents the distribution of hit rates across all users. We perform separate predictions for replay, premium live and normal live streams. We then split these users into the two categories of regularity: $0 < RV \leq thresh$ and $RV > thresh$. Unsurprisingly, the more regularly the user uploads, the higher the hit rate is. The predictions are most accurate for replay video uploading, where the hit rates are as high as 0.75 (3x the accuracy in the cases of live uploads). This is likely because it depends solely on motion, rather than user viewing behavior. This confirms that, particularly for heavy users, motion-triggered uploads *do* have the capacity to allow attackers to predict future activity. This could be an effective tool for identifying the best time for physical attacks.

4.4 Rate Change Risk

Finally, we explore the potential to identify activity changes on a camera feed via bit rate monitoring, *e.g.* identifying a person shifting from sitting to walking. This is possible due to the use of Variable Bit Rate (VBR) encoding, in which video artifacts may manifest themselves as rate changes.

Methodology: We take inspiration from Li *et al.* [31], who proved it possible to identify activities by monitoring the bit rate feed from a video stream. Their approach involves first identifying video segments (via change points), and extracting key features. By manually labeling each segment with their associated activities (*e.g.* eating, dressing, styling hair), they then train classifiers to identify activities in other feeds. With these results in mind, we next test the number of potential activity segments that can be extracted from the video streams in our dataset (1 segment maps to one activity [31]). Although we cannot associate these segments with their respective labels (*e.g.* eating), this does offer an upper-bound on how many activities could be extracted. To do this, we convert all streams into a bit rate time series, and then utilize Bayesian Online Change Point Detection (BOCPD) [12] to identify each segment in a camera's feed.

BOCPD assumes that a sequence of observations $(x_1, x_2, ..., x_t)$ contains several non-overlapping partitions ρ [19]. For a given time series, BOCPD computes the *run length*, which represents the number of time steps since the last detected change point (denoted as r_t at time t). The probability distribution of r_t can be computed using a recursive algorithm as follow:

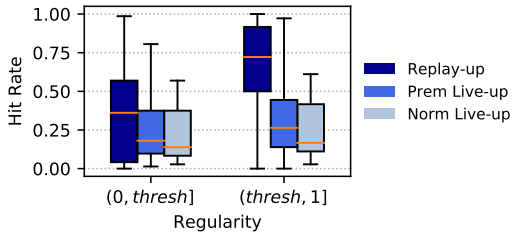


Fig. 12: Distribution of accuracy of upload behavior prediction (hit rate) for users of different ranges of regularity value.

$$P(r_t|x_{1:t}) = \frac{\sum_{r_{t-1}} P(r_t|r_{t-1})P(x_t|r_{t-1}, x_t^{(r)})P(r_{t-1}, x_{1:t-1})}{\sum_{r_t} P(r_t, x_{1:t})} \quad (2)$$

where $x_t^{(r)}$ indicates the set of observations associated with the run r_t. $P(r_t|r_{t-1})$, $P(x_t|r_{t-1}, x_t^{(r)})$ and $P(r_{t-1}, x_{1:t-1})$ are prior, likelihood, and recursive components of the above formulation. The conditional prior has non zero mass under only 2 circumstances:

$$P(r_t|r_{t-1}) = \begin{cases} H(r_{t-1}+1) & if\ r_t = 0 \\ 1 - H(r_{t-1}+1) & if\ r_t = r_{t-1}+1 \\ 0 & otherwise \end{cases} \quad (3)$$

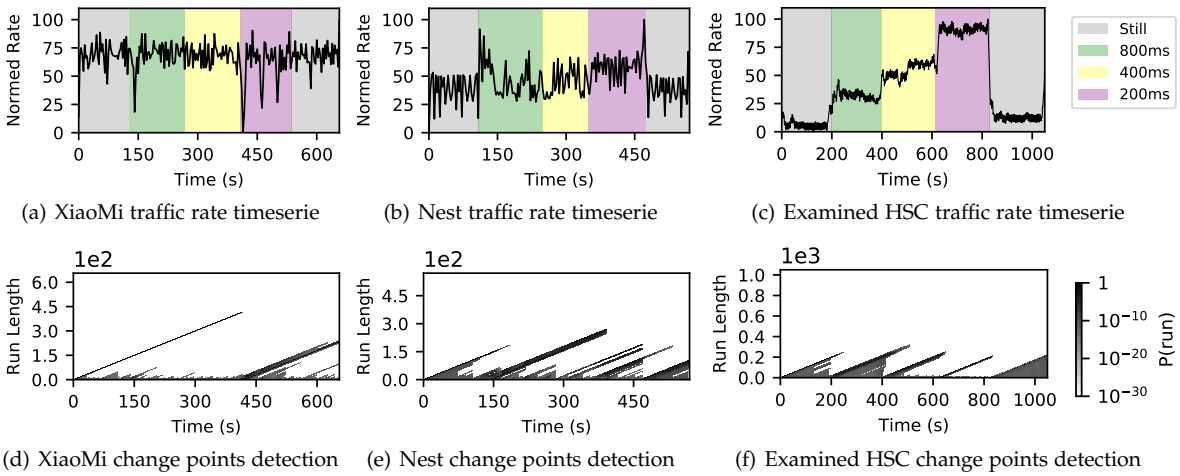
$$H(\tau) = \frac{P_{gap}(g=\tau)}{\sum_{t=\tau}^{\infty} P_{gap}(g=t)} \quad (4)$$

In the above, $H(\tau)$ is the *hazard function* [26]. The likelihood term $P(x_t|r_{t-1}, x_t^{(r)})$ therefore represents the probability that the most recent datum belongs to the current run.

Adams and MacKay [12] do not specify an exact method to identify change points after calculating the run length distributions. Thus, we propose a statistical way to identify change points: after obtaining all the run length distributions, we fit all probability values $P(r_t = 0), t \in \mathbb{N}$ into a Gaussian distribution. Since for all Gaussian distributions, 95% of the area is within 1.96 standard deviations (σ) plus the mean (μ) [13], we label any time step t as a change point when $P(r_t = 0) > \mu + 1.96\sigma$ holds. One advantage of BOCPD is that it is effective in cases where an attacker only has access to periodic (smoothed) bit rate samples (*e.g.* every 30 seconds). In such cases, each sample x can be expanded across the time period by a Poisson distribution of $\lambda = x$, namely $P(\lambda = x)$.

For context, Figure 13 highlights the outcome of this change point detection process. Here, we setup a controlled experiment, in which we connect an examined HSC, a XiaoMi HSC, as well as a Nest HSC to a Wi-Fi hotspot on a laptop. We then set those cameras in live streaming mode. Then, we present them with an animated GIF that flickers between all black and all white at different intervals (still, 800ms, 400ms, and 200ms), and meanwhile, we monitor the network traffic rates of cameras in the laptop via Wireshark.

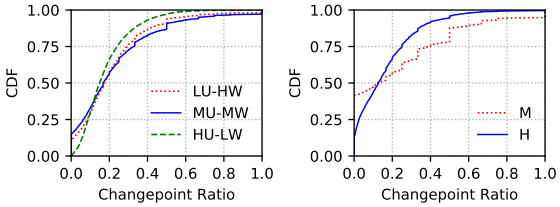
Figure 13(c) presents the bit rate time series for the examined HSC. By studying the bitrate, we see that the HSC traffic rate is sensitive to the intensity of the GIF. Figure 13(f) also shows the extract change points. This confirms the correct calculation of the run length. The same observation holds for Nest (see Figure 13(b) and 13(e)). This, however, does not hold for XiaoMi HSC (presented in Figure 13(a) and 13(d)), where there is no significant correlation between traffic rate and GIF shifting period. This is because XiaoMi does not use a susceptible encoding scheme. Note that it is out-of-scope to perform the mapping between these runs and their underlying activities, as it is necessary for an attacker to first collect ground truth mapping data for training purposes [31]. Thus, we emphasize that such an attack could only be realised by a highly equipped adversary with the ability to contextualize the segments.



(a) XiaoMi traffic rate timeserie (b) Nest traffic rate timeserie (c) Examined HSC traffic rate timeserie

(d) XiaoMi change points detection (e) Nest change points detection (f) Examined HSC change points detection

Fig. 13: Traffic rate changes of cameras when GIFs shifting period changes and the corresponding probability of run length.



(a) Premium user (b) Normal user

Fig. 14: Distribution of users' change point ratio.

Risk Exploration: First, to gain an idea of the number of *potential* activities that can be extracted from a camera feed, we measure the number of BOCPD segments in each camera's stream. To this end, we define the *Change Point Ratio* for a stream as:

$$R_u = \frac{1}{n}\sum_{i=0}^{n}\frac{C_i}{P_i} \tag{5}$$

where P_i and C_i are the number of data points and the number of identified change points in stream i's rate timeserie respectively; n is the number of up-streams generated by user u.

We report the distribution of change point ratios for each user in Figure 14, where we again separate users into their categories as identified via our earlier GMM clustering. The change point ratio distribution of the 3 groups of premium users is quite close. The most noteworthy is that under 1% of HU-LW users have never had a change point (this is about 10% in the other two relatively inactive uploading premium user groups). This indicates that the remaining majority of users *do* have activity changes within the streams. Compared with premium users, more normal users have a zero change point ratio (40% medium normal users and 20% high normal users), implying very few activity changes in their streams. This distinct behavior is partly because only live streaming mode is available for normal users, which results in less regular activity than motion-triggered capture. Nevertheless, with appropriate training data, this implies that the attack detailed in Li *et al.* [31] would have widespread applicability.

Activity Switch Patterns: We are also interested in users' activity switch patterns (*ASP*), since people often do things in a certain logical order (*e.g.* washing their hands before dinner). These sequences could therefore add context to any inferences performed by an attacker. Unlike the controlled experiment in a laboratory [31], our dataset is composed of real-user traces and contains complex activities that are not exhaustive. We first divide camera traffic time series into *activity segments*, where two consecutive activity segments are separated by a change point that is identified using the above BOCPD-based methodology. We then cluster activity segments so that the activities in the same group have similar characteristics, and finally we study the ASP based on activity categories.

To this end, we first extract 7 features for activity segments, and then input the features into the unsupervised learning algorithm (K-Means) to obtain clustering results. Specially, we extract features from the perspectives of both the time domain and the frequency domain:

1) *The duration of the segment:* The duration (d) reflects the **length** of the activity.
2) *The mean bit rate:* The mean bit rate (b_{avg}) can show the **intensity** of the activity.
3) *The median bit rate:* To avoid the effect of the churn, we also take the median (b_{mid}) into consideration.
4) *The variance of the bit rate:* Through the variance (var) we can understand how **variable** is the activity.
5) *The skewness of the bit rate:* Skewness represents the **degree of asymmetry** in the data distribution. Considering a segment $S = (b_1, b_2, ..., b_i, ..., b_n)$, b_{avg} is the average of the bit rate, the skewness can be defined as follow:

$$skewness = \frac{\frac{1}{n}\sum_{i=1}^{n}(b_i - b_{avg})^3}{(\frac{1}{n-1}\sum_{i=1}^{n}(b_i - b_{avg})^2)^{3/2}} \tag{6}$$

6) *The kurtosis of the bit rate:* Kurtosis exhibits the **sharpness** of the peak of the distribution curve. Considering a segment $S = (b_1, b_2, ..., b_i, ..., b_n)$, b_{avg} is the average of all bit rate, the kurtosis can be defined as follow:

$$kurtosis = \frac{\frac{1}{n}\sum_{i=1}^{n}(b_i - b_{avg})^4}{(\frac{1}{n}\sum_{i=1}^{n}(b_i - b_{avg})^2)^2} - 3 \tag{7}$$

TABLE 4: Mean value of major features for each segment cluster.

no.	d (s)	b_{avg} (kbps)	%	feature
#1	33.6	376.8	43.9	light intensity
#2	35.7	905.2	27.0	high intensity
#3	279.9	888.7	21.1	long-lasting
#4	105.6	924.3	8.0	burst motion

7) *The first k Fast Fourier Transform (FFT) coefficients*: To discover the **periodic characteristics** of the activity, we also take the frequency domain into consideration: We apply FFT to each segment and use the first k coefficients as the features[6].

Recall that the replay uploading is triggered by the detected activity. Therefore, we focus on the replay-up streams from the MU-MW and HU-LW premium users. To avoid possible bias caused by stream length, we only consider replay up streams of duration between 2 and 30 minutes. This is because such streams are not too short to contain many meaningless random events (*e.g.* , someone shortly passing by), Similarly, it is long enough to cover a complete activity.

After extracting the features from above segments, we apply Z-Score and Principal Component Analysis (PCA) for preprocessing. Then we input the outcome to K-Means algorithm for clustering. We experiment with k from 2 to 10, and determine $k = 4$ based on the relatively small Davies Bouldin Index (DBI) [20]. We show the mean value of some distinctive features (duration d and average bit rate b_{avg}) for each cluster in Table 4. Note that the activity segments in a group exhibit similar traffic rate pattern, but may have different types of activities. For instance, the segments of chewing food at home and typing keyboard in an office may be classified into one group, because both are of light intensity. We can summarize the characteristics of the four categories of activities as follows: The cluster 1 and cluster 2 are the two major activity types ($> 70\%$ of activity segments). The streams in both categories are short in duration, but differ from each other in bit rate level. It seems that the activities in the cluster 1 are triggered by movements of light intensity, resulting in the lowest bitrate. On the other hand, the second highest bit rate implies that activities in the cluster 2 are triggered by movements of high intensity. Thus, we term the cluster 1 and 2 as *light intensity* and *high intensity*, respectively. The cluster 3 accounts for about 1/5 of all activities, and activities in this group last for the longest duration. We term this group as *long-lasting* activities. Finally, the least populated cluster 4 shows the highest average bit rate, which potentially results from sudden movements. Thus, we term it as *burst motion*.

Next, we examine the activity switch pattern by examining the probability of one type of activity (A_1) switching to another (A_2) $Prob\{A_1 \rightarrow A_2\}$, where A_1 and A_2 are two consecutive activity segments separated by a change point. Specially, we compute $Prob\{A_1 \rightarrow A_2\} = n_{A_1 \rightarrow A_2}/N \times 100\%$, where N is the total number of activity switches observed in our dataset, and $n_{A_1 \rightarrow A_2}$ is the number of switches from A_1 to A_2. The results are shown in Figure 15, where the number marked on the grid x, y shows $Prob\{y \rightarrow x\}$.

6. We set k to empirical value 5.

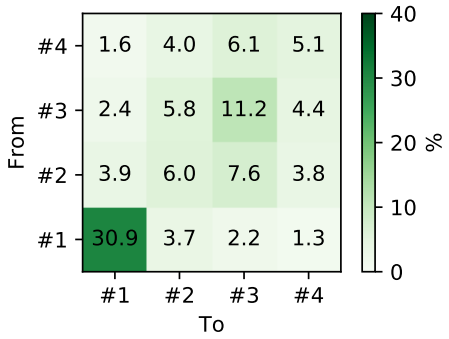


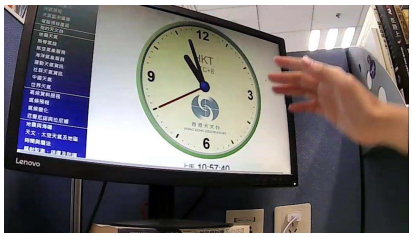
Fig. 15: Percentage of the types of two activities that occur in succession. The number indicates the percentage for switching from type Y to type X. All numbers sum up to 100%

We have following observations. First of all, the most common switch pattern is the switching between activities of the same type: The proportion of occurrences of such switches is 52.3%, and the probability of switching from one light-intensity to another light-intensity ($1 \rightarrow 1$) is as high as 30.9%. Secondly, the possibility of switching from (to resp.) the activity of cluster 1 to (from resp.) others is lower. Recall that, the activities in cluster 1 are far less intensive than those in other clusters, as indicated by the very low bit rates. We conjecture this observation is because it is relatively infrequent that people suddenly change from low-intensive activities to high-intensive ones. Finally, we find that a long-lasting activity (cluster 3) is usually followed by another long-lasting activity.

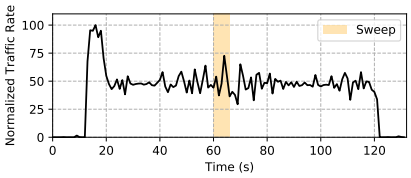
Although, we cannot link the activity change patterns into specific activities, we make the first attempt to show the possibility of linking the changes to users' activities, and shed light on future work. Furthermore, our taxonomy could be used by attackers to infer a general category of activity taking place on a camera feed.

4.5 Toward Privacy Risk Mitigation

The root cause of the three risks is that there is a correspondence between the traffic rate and the working state of the camera. Our results show that premium users are most at risk, which is attributed to two factors: (*i*)the heavier usage of live streaming by the premium users means a higher possibility of exposure to the risks; (*ii*)the exclusive replay-up mode further increases the risk possibility. It is therefore necessary to alleviate this correspondence. In the simplest case, this could be done by artificially triggering camera activity to introduce noise to any inferences, for example, by the user directly placing a moving object (*e.g.* clock) in front of the camera for motion-triggered recording. To examine the efficacy of this strawman solution, we setup a controlled experiment, where a clock is set in front of the examined HSC (about 1 metre away). We observed motion detection recording being triggered immediately. This strategy can also cover up the rate change caused by the movement of light intensity: We swept hand for about 5 seconds (Figure 16(a)), and we observed no significant HSC traffic rate change (Figure 16(b)). Indeed, the movement of the second hand in the clock adds noise to the traffic rate.



(a) The screenshot of the recording as the hand swept.



(b) Traffic rate of the HSC.

Fig. 16: No obvious HSC traffic rate change was observed when a hand swept past the clock background.

This, however, is undesirable for several reasons, not least because it would waste resources: in our controlled experiments, it generates over 50MB of wasted traffic per hour. Another reason for not preferring this strawman solution is that the moving object (here are the hands in the clock) should not be far from the camera. Otherwise, the noise would be too low to undermine the attacks. A close moving object before the camera would limit its coverage, even make it useless.

Thus, a superior option would be for each HSC to randomly generate streams, thereby undermining the attacks. Such streams could be tagged in order to inform the server to discard them. Notably, the times, duration and traffic rate pattern must be random. HSCs could also perform traffic shaping to flatten spikes in the bit rate [16], [48]. Note that half of the users wait at least 10 minutes before viewing newly uploaded replay videos, suggesting that such traffic shaping could be easily performed without an adverse impact on user experience.

4.6 Discussion

It should be noted that the traffic surge risk and regularity risk are not specific for HSCs. For instance, Apthorpe *et al.* [18] showed that the traffic rates of sleep sensors can be exploited to infer when the user falls asleep and when to get up. However, the (traffic) rate change risk is unique to HSC due to the use of VBR encoding for video compression, where the bit rate of a video stream is closely related to the video content.

Indeed, the HSC traffic raises unique challenges and causes substantial impacts: (*i*)HSCs are always-on sensors installed mostly in intimate places for monitoring. The replay mode available for premium users even automatically uploads recordings of the activities in front of the cameras. These features make HSCs much more sensitive to privacy risks; (*ii*)the rate change risk means that a sophisticated attacker can even infer the activities happening in the user's home by simply monitoring the (encrypted) HSC traffic; (*iii*)the risks caused by the HSC traffic are difficult to address because obfuscating the traffic rate changes may introduce much traffic overhead.

Our proposed solutions in Section 4.5 are the first attempt toward the privacy risk mitigation for HSCs. These solutions, however, require further validation and evaluation before wide adoption. We leave more sophisticated solutions as future work.

5 RELATED WORK

Privacy Leakages from IoT Traffic: In spite of encryption in IoT, several recent studies have shown the possibility of privacy leakages from application traffic. Ren *et al.* [40] characterised the information exposure for 81 IoT devices from four major perspectives. Li *et al.* [31] showed the possibility of inferring Activities of Daily Living from encrypted surveillance video traffic. Apthorpe *et al.* [18] showed that attackers can infer users' activities from a set of IoT devices' (including Nest HSC) encrypted traffic. Wood *et al.* [46] investigated medical IoT devices that may reveal sensitive medical conditions and behaviors. Copos *et al.* [24] presented a scheme that could infer whether a home is occupied by parsing characteristics of the network traffic from smart thermostats. Xu *et al.* [30] devised a system to separate the wireless camera stream from the others and infer the presence of person in the house by inspecting the camera stream. They further use a similar scheme to find hidden camera [23]. Srinivasan *et al.* [43] presented a novel attack in home IoT systems, by which an attacker can observe home private activities. They then proposed and evaluated a set of privacy-preserving design guidelines for future wireless IoT systems. Other works examined the vulnerability of unprotected Internet cameras. Xu *et al.* [47] investigated IP cameras without password protection in *insecam* [7] to examine the vulnerabilities of those cameras. Song *et al.* [42] conducted a systematic study on live webcams from both aggregation sites and individual webcams, and analysed the spatial and content features of live webcams. Our work differs from the above in two aspects: (*i*) These studies mainly rely on active measurements with a limited number of cameras, whereas our work leverages large-scale service logs of over 200K cameras from a major HSC service provider. We also conduct active measurements using three different types of HSCs to complement the passive logs. (*ii*) While most of above works only examined one privacy risk, we investigate three privacy risks.

To address the possible privacy leakages in IoT applications, several countermeasures have been proposed. Apthorpe *et al.* [17] [16] proposed strategies, including traffic shaping and tunneling to protect IoT device consumers from side-channel traffic rate privacy threats. Zhang *et al.* [48] proposed to reshape packet features through dynamically scheduling packets over multiple virtual MAC interfaces, in order to obscure the features of the original traffic. The Replacement AutoEncoder [37], on the other hand, transforms discriminative features contained in time series data to the features that are common in non-sensitive inferences, to protect users' privacy. Some of these solutions, like traffic shaping, can also be applied to HSCs to preserve privacy.

User Behavior Inspection: To the best of our knowledge, we conduct the first measurement work on the HSC user behavioral pattern based on a large-scale passive log dataset. Nevertheless, the user behavioral patterns of other Internet video applications have been examined in various works. Li *et al.* [35] examined user behavior patterns of a mobile live streaming service and identified the traffic waste due to the less frequent access to the uploaded content. Raman *et al.* [39] explored a long period of *Facebook Live* data and found that most of the broadcasts are short and go unwatched which introduces unnecessary network burden. Our results also have similarities with mobile personal livecast systems [36], [45], which also exhibit short stream characteristics. Brodersen *et al.* [22] investigated the relationship between popularity and locality of online YouTube videos by examining whether YouTube videos exhibit geographic locality of interest. Deng *et al.* [25] explored the unique nature of the Twitch platform, and found that Twitch is very different from existing video platforms. The HSC services differ from other Internet video applications in that: (*i*) the content is more private; (*ii*) HSCs often provide live streaming mode, as well as motion detection mode; (*iii*) HSCs are unicast in nature, where content is only available to the owner of the camera. These features yield the distinct behavior patterns of HSCs as we found in this paper.

6 Conclusion

In this paper, we have presented a large-scale measurement work of a major HSC system, highlighting several novel findings. For instance, only the top 5% of cameras (largely motion-triggered uploads) produce about 95% of replay upload traffic. Such skewed workload results in a significant fraction of content going unwatched and therefore wasting resources. These previously unknown patterns contribute to the growing body of work focused on optimizing home IoT devices. We also inspected the privacy implications of using HSCs, driven by the close integration between real-world activities and subsequent network traffic. We disclosed a range of privacy inferences, and have offered an upper bound for the predictability of user patterns. The susceptibilities of users to these risks differ, and we identified a subset of heavy users who are most vulnerable. Although our passive log dataset comes from one HSC service, we test the privacy risks in two other popular HSCs and note the privacy attacks are equally applicable to most other HSC brands, due to their use of Variable Bit Rate encoding.

Acknowledgment

This work was supported in part by National Key R&D Program of China: 2019YFB1802800, the NSF of China (61725206 and 62072437), the Youth Innovation Promotion Association CAS, and the Project "PCL Future Greater-Bay Area Network Facilities for Large-scale Experiments and Applications (LZC0019)".

References

[1] Hacked cameras, dvrs powered today's massive internet outage. https://krebsonsecurity.com/2016/10/hacked-cameras-dvrs-powered-todays-massive-internet-outage/, 2016.

[2] Crypto-pan. https://www.cc.gatech.edu/computing/Networking/projects/cryptopan/, 2018.

[3] Exponentialsmoothing in statsmodels's documentation. https://www.statsmodels.org/dev/generated/statsmodels.tsa.holtwinters.ExponentialSmoothing.html, 2018.

[4] Fit gaussian mixture model to data - matlab fitgmdist. https://ww2.mathworks.cn/help/stats/fitgmdist.html, 2018.

[5] Hikvision ezviz camera. https://www.ys7.com/, 2018.

[6] Home security camera market research report- global forecast 2023. https://www.marketresearchfuture.com/reports/home-security-camera-market-3787, 2018.

[7] Insecam - world biggest online cameras directory. http://www.insecam.org/, 2018.

[8] Nest cam indoor. https://nest.com/cameras/nest-cam-indoor/overview/, 2018.

[9] Netgear arlo camera. https://www.arlo.com/en-us/, 2018.

[10] Team cymru. https://www.team-cymru.com/, 2018.

[11] Xiaomi smart camera. https://www.mi.com/micamera/, 2018.

[12] R. P. Adams and D. J. MacKay. Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*, 2007.

[13] D. G. Altman and J. M. Bland. Standard deviations and standard errors. *Bmj*, 331(7521):903, 2005.

[14] S. Aneja, N. Aneja, and S. Islam. Iot device fingerprint using deep learning. *arXiv: Networking and Internet Architecture*, 2019.

[15] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou. Understanding the mirai botnet. In *SEC'17 Proceedings of the 26th USENIX Conference on Security Symposium*, pages 1093–1110, 2017.

[16] N. Apthorpe, D. Y. Huang, D. Reisman, A. Narayanan, and N. Feamster. Keeping the smart home private with smart(er) iot traffic shaping. *CoRR*, abs/1812.00955, 2018.

[17] N. Apthorpe, D. Reisman, and N. Feamster. Closing the blinds: Four strategies for protecting smart home privacy from network observers. *arXiv preprint arXiv:1705.06809*, 2017.

[18] N. Apthorpe, D. Reisman, and N. Feamster. A smart home is no castle: Privacy vulnerabilities of encrypted iot traffic. *CoRR*, abs/1705.06805, 2017.

[19] D. Barry and J. A. Hartigan. Product partition models for change point problems. *Ann. Statist.*, 20(1):260–279, 03 1992.

[20] J. C. Bezdek and N. R. Pal. Some new indexes of cluster validity. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 28(3):301–315, 1998.

[21] T. Böttger, F. Cuadrado, G. Tyson, I. Castro, and S. Uhlig. Open connect everywhere: A glimpse at the internet ecosystem through the lens of the netflix cdn. *ACM SIGCOMM Computer Communication Review*, 48(1):28–34, 2018.

[22] A. Brodersen, S. Scellato, and M. Wattenhofer. Youtube around the world: geographic popularity of videos. In *Proceedings of the 21st international conference on World Wide Web*, pages 241–250. ACM, 2012.

[23] Y. Cheng, X. Ji, T. Lu, and W. Xu. On detecting hidden wireless cameras: A traffic pattern-based approach. *IEEE Transactions on Mobile Computing*, 19(4):907–921, 2020.

[24] B. Copos, K. N. Levitt, M. Bishop, and J. Rowe. Is anybody home? inferring activity from smart home network traffic. In *2016 IEEE Security and Privacy Workshops (SPW)*, pages 245–251, 2016.

[25] J. Deng, F. Cuadrado, G. Tyson, and S. Uhlig. Behind the game: Exploring the twitch streaming platform. In *2015 International Workshop on Network and Systems Support for Games (NetGames)*, pages 1–6. IEEE, 2015.

[26] M. Evans, N. Hastings, and B. Peacock. *Statistical Distributions*. Wiley Series in Probability and Statistics. Wiley, 2000.

[27] X. Feng, Q. Li, H. Wang, and L. Sun. Acquisitional rule-based engine for discovering internet-of-things devices. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 327–341, Baltimore, MD, 2018. USENIX Association.

[28] O. L. Haimson and J. C. Tang. What makes live events engaging on facebook live, periscope, and snapchat. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pages 48–60. ACM, 2017.

[29] A. Halfaker, O. Keyes, D. Kluver, J. Thebault-Spieker, T. T. Nguyen, K. Shores, A. Uduwage, and M. Warncke-Wang. User session identification based on strong regularities in inter-activity time. In *Proceedings of the 24th International Conference on World Wide Web*, pages 410–418, 2015.

[30] X. Ji, Y. Cheng, W. Xu, and X. Zhou. User presence inference via encrypted traffic of wireless camera in smart homes. *Security and Communication Networks*, 2018:1–10, 2018.

[31] H. Li, Y. He, L. Sun, X. Cheng, and J. Yu. Side-channel information leakage of encrypted video stream in video surveillance systems. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, 2016.

[32] J. Li, Z. Li, G. Tyson, and G. Xie. Your privilege gives your privacy away: An analysis of a home security camera service. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020.

[33] Z. Li, M. A. Kaafar, K. Salamatian, and G. Xie. Characterizing and modeling user behavior in a large-scale mobile live streaming system. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2675–2686, 2017.

[34] Z. Li, J. Lin, M.-I. Akodjenou, G. Xie, M. A. Kaafar, Y. Jin, and G. Peng. Watching videos from everywhere: a study of the pptv mobile vod system. In *Proceedings of the 2012 Internet Measurement Conference on*, pages 185–198, 2012.

[35] Z. Li, X. Wang, N. Huang, M. A. Kaafar, Z. Li, J. Zhou, G. Xie, and P. Steenkiste. An empirical analysis of a large-scale mobile cloud storage service. In *Proceedings of the 2016 ACM Internet Measurement Conference*, IMC '16, page 287–301, 2016.

[36] M. Ma, L. Zhang, J. Liu, Z. Wang, H. Pang, L. Sun, W. Li, G. Hou, and K. Chu. Characterizing user behaviors in mobile personal livecast: Towards an edge computing-assisted paradigm. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 14(3):66, 2018.

[37] M. Malekzadeh, R. G. Clegg, and H. Haddadi. Replacement autoencoder: A privacy-preserving algorithm for sensory data analysis. In *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 165–176, 2018.

[38] A. J. Pinheiro, J. de M. Bezerra, C. A. Burgardt, and D. R. Campelo. Identifying iot devices and events based on packet length from encrypted traffic. *Computer Communications*, 144:8–17, 2019.

[39] A. Raman, G. Tyson, and N. Sastry. Facebook (a) live?: Are live social broadcasts really broadcasts? In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 1491–1500. International World Wide Web Conferences Steering Committee, 2018.

[40] J. Ren, D. J. Dubois, D. Choffnes, A. M. Mandalari, R. Kolcun, and H. Haddadi. Information exposure from consumer iot devices: A multidimensional, network-informed measurement approach. In *Proceedings of the Internet Measurement Conference*, 2019.

[41] A. Sivanathan, H. H. Gharakheili, F. Loi, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman. Classifying iot devices in smart environments using network traffic characteristics. *IEEE Transactions on Mobile Computing*, 18(8):1745–1759, 2019.

[42] J. Song, Q. Li, H. Wang, and L. Sun. Under the concealing surface: Detecting and understanding live webcams in the wild. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 4:1 – 25, 2020.

[43] V. Srinivasan, J. A. Stankovic, and K. Whitehouse. Protecting your daily in-home activity information from a wireless snooping attack. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 202–211, 2008.

[44] D. Stohr, T. Li, S. Wilk, S. Santini, and W. Effelsberg. An analysis of the younow live streaming platform. In *2015 IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops)*, pages 673–679, Oct 2015.

[45] B. Wang, X. Zhang, G. Wang, H. Zheng, and B. Y. Zhao. Anatomy of a personalized livestreaming system. In *Proceedings of the 2016 Internet Measurement Conference*, pages 485–498. ACM, 2016.

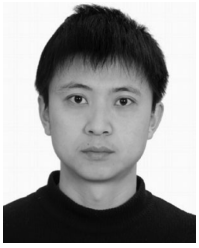
[46] D. Wood, N. Apthorpe, and N. Feamster. Cleartext data transmissions in consumer iot medical devices. In *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy*, pages 7–12, 2017.

[47] H. Xu, F. Xu, and B. Chen. Internet protocol cameras with no password protection: An empirical investigation. In R. Beverly, G. Smaragdakis, and A. Feldmann, editors, *Passive and Active Measurement*, pages 47–59, Cham, 2018. Springer International Publishing.

[48] F. Zhang, W. He, and X. Liu. Defending against traffic analysis in wireless networks through traffic reshaping. In *2011 31st International Conference on Distributed Computing Systems*, pages 593–602, 2011.

[49] S. Zheng, N. Apthorpe, M. Chetty, and N. Feamster. User perceptions of smart home iot privacy. *arXiv: Human-Computer Interaction*, 2:200, 2018.

Jinyang Li is currently a Ph.D candidate at the Institute of Computing Technology (ICT), Chinese Academy Sciences (CAS). He received his B.S. degree in Computer Science from Sichuan University, Chengdu, China, in 2017. His research interests include IoT security, live streaming, and Internet measurement.



Zhenyu Li received the BS degree from Nankai University in 2003 and the PhD degree in Graduate School of Chinese Academy of Sciences (CAS) in 2009. He is a professor at the Institute of Computing Technology, CAS. His research interests include Internet measurement and Networked Systems.

Gareth Tyson received the PhD degree from Lancaster University in 2010. He is a senior lecturer at Queen Mary University of London. His research interests include Internet measurements, content distribution, and the future Internet.

Gaogang Xie received the PhD degree in computer science from Hunan University, in 2002. He is a professor in the Computer Network Information Center, Chinese Academy of Sciences. His research interests include Internet architecture, SDN/NFV, and Internet measurement.