IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received August 3, 2018, accepted September 2, 2018, date of publication September 13, 2018, date of current version October 12, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2869848

Performance Analysis of Personal Cloud Storage Services for Mobile Multimedia Health Record Management

MAHMUDA AKTER[1], ABDULLAH GANI[1], MD. OBAIDUR RAHMAN[2], (Member, IEEE), MOHAMMAD MEHEDI HASSAN[3], (Senior Member, IEEE), AHMAD ALMOGREN[3], AND SHAFIQ AHMAD[4]

[1]Mobile Cloud Computing Research Lab, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia
[2]Department of Computer Science and Engineering, Dhaka University of Engineering & Technology, Gazipur 1700, Bangladesh
[3]College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
[4]Department of Industrial Engineering, College of Engineering, King Saud University, Riyadh 11421, Saudi Arabia

Corresponding author: Mohammad Mehedi Hassan (mmhassan@ksu.edu.sa)

This work was supported by the Deanship of Scientific Research, King Saud University, through Research Group under Grant RGP-1437-35. The authors also thank the DSR and RSSU at King Saud University for their technical support.

ABSTRACT Recently, the trend of mobile multimedia services and applications being used for e-health is growing in popularity. This is because people can get access to their electronic personal health records (PHRs), such as medical history, lab reports from an X-ray, MRI, clinical audio-visual notes, EEG/ECG data, and insurance policy details from anywhere, at any time, from their mobile or handheld devices. In this scenario, a medical care provider or a patient is responsible for uploading and managing the patient's health information via cloud storage services. There are a number of personal cloud storage services that could be used such as Dropbox, Google Drive, OneDrive, and Box. However, the different designs of these personal cloud storage services mean there are differences in their performance in terms of storing and managing PHRs. In this paper, we present the details of our study on the performance of personal cloud storage services, and we highlight the strengths and weaknesses of such services in terms of PHR management. We investigate the performance of personal cloud storage services by conducting a qualitative and quantitative analysis of them. The qualitative analysis highlights strengths and weaknesses in terms of supported capabilities/features and shortcomings in terms of potential features that have not been implemented. The capabilities we analyze are chunking, bundling, deduplication, delta-encoding, and data compression. In the quantitative analysis, we investigate performance in terms of control data overhead, impact of data size on number of packets as well as transmission rate, synchronization initialization time, and protocol overhead. During testing with diverse benchmark size on distinct cloud storage services, we attained an average transmission of 93%, 3%, and 4% for application data, control data, and other data, respectively. This research allows us to identify open issues and to determine future directions for developing an efficient personal cloud storage service.

INDEX TERMS Mobile multimedia health record, personal cloud storage, performance comparison, Dropbox, Google drive.

I. INTRODUCTION

In recent years, personal cloud storage (PCS) services are more commonly being used for storing and managing electronic Personal Health Records (PHRs) of users/patients. PCS allows users/patients to access their PHRs from anywhere, at any time, from their mobile devices, and it permits the storage, retrieval, sharing, and synchronization of PHR files in cloud servers [1]–[3]. Nowadays, PCS is popular because companies are offering significant amounts of remote storage at little or no costs [4]. More individuals are being attracted by these offers, which allow them to save personal documents, synchronize devices, and share content with great simplicity. Online service providers are being pushed to enter the cloud storage market due to high public interest. Services such as Dropbox, OneDrive, Google Drive, and Box are becoming pervasive in individuals' routines. Such applications are data-intensive and their increasing usage already produces a remarkable share of Internet traffic [5].

These PCS services provide users with very convenient tools, especially given the increasing diversity of user devices requiring synchronization. With such resources and tools, mostly available for free, users are likely to upload ever larger amounts of personal and private data [6]. The number of registered PCS users will increase on a daily basis, as claimed by the Dropbox, Google Drive, OneDrive and Box, who have 400 million, 240 million, 250 million, and 32 million members, respectively. The main challenges faced by PCS services are information security/unauthorized data access, registration errors, and file synchronization. Security can be vulnerable because cloud providers or their representatives may be malicious, negligent, or even compromised by a third party [7]. Registration techniques usually yield registration errors, which must be dealt with to ensure the relevance of factors such as apps, storage availability, features, and collaboration [8]. File synchronization may occur due to tight resource limitations, suffer from temporary outages or even shutdowns, and sometimes silently corrupt or leak user data [9]. While being resilient to these challenges, the features of a representation shall also enable to meet user's expectations [10], [11].

In this paper, we investigate PCS services in detail by analyzing their strengths and weaknesses for handling mobile multimedia healthcare data. our contributions are summarized as follows:

- We explore the performance of personal cloud storage services by conducting a qualitative and quantitative analysis of them for media healthcare data record management
- In the qualitative analysis, we investigate the capabilities of chunking, bundling, deduplication, delta-encoding, and data compression.
- In the quantitative analysis, we investigate performance in terms of control data overhead, application data exchanged, impact of data size on number of packets, impact of data size on transmission rate, synchronization initialization time, and protocol overhead.
- The cumulative research on individual PCS services has shown the significance of Dropbox compared with the other services reviewed in this study.
- Finally, we discuss open issues and list potential future directions.

Finally, we discuss open issues and list potential future directions. The remainder of this paper is organized as follows. In Section 2, a literature review is first presented in order to highlight similar efforts in this domain. Individual storage software platforms are described in Section 3. In Section 4, a qualitative analysis of PCS services is presented, and distinct PCS capabilities are highlighted. Section 5 evaluates the performance of distinct PCS services for different data sizes through traffic analysis and measuring transmission rate, synchronization time, throughput, and protocol overhead. Concluding remarks and suggestions for future work are presented in Section 6.

II. LITERATURE REVIEW

PCS has recently been gaining popularity in the arena of cloud computing, which empowers clouds with file synchronization, security, and availability services [12]. A personal cloud is a collection of virtual machines that share computational resources running on unused PCs at the edge. A recent trend [13] defines a personal cloud as a cloud operating system that offers a core set of services around identity, trust, data access, and even programming models. In this regard, the term ''personal cloud'' has gained much attention, with recent research focusing on it [14], [15]. These papers relate the term ''personal cloud'' to online cloud storage services, such as Dropbox, Google Drive, OneDrive, Box, Kingsoft, Wuala, and Amazon. In general, there are two major distinguishing features of PCS: representation and synchronization. In this paper, we focus on the PCS platforms that deal with data syncing and sharing from heterogeneous devices.

A. PERSONAL CLOUD STORAGE

Cloud services have changed the way computing power is delivered to customers, by offering on-demand computing and storage capacity in remote data centers. Cloud computing and, in turn, cloud services have been interpreted and compared in several manners [16]. For example, the services offered by cloud providers have been categorized according to what is delivered (e.g., infrastructure, platform, or software), their deployment model (e.g., public, private or hybrid), and other factors. One reference architecture for cloud storage federation service implementation is proposed in [17].

In recent years, many organizations have begun to consider migrating their services to the cloud [18]. However, in spite of huge public demand, little research has been conducted on PCS design and performance [19]–[22]. A detailed comparison of different PCS service providers has been conducted in studies [23]–[26]. Beside performance evaluation [27], [28], other aspects of PCS services, such as the nature of data center traffic [29], quality of experience [30], and service variability [31], have also been considered for current PCS designs and implementations. Specifically, a complete configuration as well as a systematic performance measurement is a current user demand. Outsourcing to the cloud is regarded as profitable due the advantages of lower costs, adaptable provisioning, and high scalability. On the contrary, this migration brings about some downsides. Cloud providers have repeatedly had significant failures reported [32]. Similarly, privacy of cloud services has been a hot issue owing to the possibility of direct access to users' private data by providers and, more alarmingly, foreign governments [33], [34]. In this regard, cloud services have been classified according to what is delivered (e.g., infrastructure, platform, or software), their deployment model (e.g., public, private, or hybrid), and other factors. Infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS) have been used to classify what distinct cloud services offer [36]–[40]. In general, XaaS [35] denotes X as a service.

Unfortunately, not all services are trustworthy or reliable in terms of security and availability. Storage services routinely lose data due to internal faults [41], [42] or bugs [43]–[45], leak users' personal data [46], [47], and alter user files by adding metadata [48]. They may also block access to content (e.g., Digital Millennium Copyright Act takedowns [49]). Occasionally, entire cloud services go out of business (e.g., Ubuntu One [50]). Security and privacy issues in PCS services for deployment of data duplication have been studied in [51] and [52].

B. SYNCHRONIZATION OF PCS

File Synchronization is the core of any personal cloud. Varieties of commercial sync protocol have been designed and implemented, along with individual PCS in recent years, due to huge public demand for online file syncing services. In [29], Schwartz *et al.* proposes a model to calculate the waiting time for PCS file synchronization services. Dropbox utilized third-party libraries, such as librsync for file synchronization; however, this library suffers from uncertainties due to the nature of the rsync algorithm [53]. Other prominent tools, such as unison [54], that use the same basic algorithm suffer from these same uncertainties. rsync is symmetric and provides pairwise synchronization between two devices, where the rsync utility running on each machine must have local access to the entire file. This prerequisite represents the first practical constraint in the adoption of rsync, because working at the file level prevents efficient data deduplication. StackSync [55], a modular and extensible open source software framework developed in Java, contains most of the software pieces to run a basic personal cloud. Specifically, for support in metadata management, efficient notification, deduplication, and data storage, StackSync is widely used. Obviously, during simultaneous changes in a file by different user, a conflict occurs in StackSync. Unfortunately, ownCloud[12] is neither modular nor an extensible framework as StackSync is. Research has focused on developing web front-end, rather than working on the internal mechanism of ownCloud. In this regard, inefficient pull strategy with huge control overhead is not adaptable. Moreover, the data flows and sync flows are tightly coupled and prohibited from integrating PCS capabilities such as chunking and deduplication mechanisms. Philip Heckel developed an open source personal cloud, called Syncany, in Java. Providing a modular and extensible mechanism, Syncany can synchronize with a variety of storage backends for PCS capabilities, such as chunking. The significant weakness observed in Syncany is lack of scalability, because of its heavy pull strategy with data flows and metadata, which are tightly coupled. Instead of starting from scratch, MetaSync [56] provides file synchronization on top of multiple existing storage providers' services, which leverages resources that are mostly well provisioned, reliable, and inexpensive. While each service provides unique features, their common purpose is to synchronize a set of user files between personal devices and the cloud. Through combining multiple providers' services, MetaSync provides users with larger storage capacity, and more importantly, a more reliable and better performing service.

III. STORAGE SOFTWARE PLATFORMS

Cloud storage is an intangible entity, which provides clients the ability to store, retrieve, share, and synchronize files on a relative core. Cloud storage supports cross-platform compatibility and scalability, which can be evaluated in terms of the abovementioned features.

A. DROPBOX

The simplest cloud storage service is Dropbox, which is also very reliable. Supporting cross-platform, the Dropbox client can access documents from websites in both its desktop applications on Linux, Windows, and Mac OS and its mobile applications on iOS, Android, BlackBerry, and Kindle Fire. Dropbox offers 2 GB of free storage to its registered users. However, users can achieve referral advantages, e.g., 500 MB of extra space for each friend they refer to Dropbox. In this way, clients can have aggregate up to 20 GB of free cloud storage, as well as the initial 2 GB storage. Although there is no file size restriction for any of the Dropbox apps, 10 GB is the maximum file size that can be uploaded using their website.

The Dropbox website supports unique files types, such as Microsoft (MS) Office, image, audio, video and Apple iWork. Dropbox does not allow online editing, which means documents must be downloaded to a device in order to edit them. Currently, Dropbox is integrated with the following features: events tracking, selective folder syncing, sharing link, version history, and social media integration. For maintaining collaboration with other Dropbox account holders, Dropbox provides the capability of sharing documents, which allows updates to be viewable by all collaborators. Even without installing the desktop client version, users can download shared documents directly from Dropbox's web interface. Both Dropbox and non-Dropbox users can access links to publicly shared documents.

B. GOOGLE DRIVE

Google Drive is accessible through iOS and Android OS mobiles and computers that run Windows or Mac OS; however, unlike Dropbox, it has no native support for Linux and relies on third party programs. Users access their Google Drive storage service via their Gmail account. A sufficient amount of free storage space is allocated to each user account (15 GB). However, additional space (ranging from 25 GB to 16 TB) can be purchased through a subscription service. The maximum file upload size is restricted to 5 TB for each client.

The Google Drive website supports unique file types, such as Photoshop (.PSD), Adobe Illustrator (.AI), Scalable Vector Graphics, and Autodesk AutoCad. In general, clients can edit MS Office documents only after converting them to a Google

TABLE 1. Simple comparison of PCS services [54].

Cloud Name	Advantages	Limitations	Significant Features
Dropbox	• Efficient cross-platform capabilities • Simple and user-friendly	File display is limited to the user	Integration with social media
Google Drive	• Easy installation and use • Easy access to documents	Automatic upload from mobile to the cloud is not available	Appeals to Google enthusiasts, or anyone who finds office tools integrated with their cloud storage useful
OneDrive	• Works seamlessly with Windows OS devices • Integrated with useful apps, such as MS Photos and MS Web Apps	Reduced functionality if not running Windows operating system and storage limit of 20,000 files	Windows PC, tablet, and mobile phone devices
Box	• Efficient for business customers due to its many tools for collaboration and its file privacy controls	Website is difficult to navigate due to its many features	Secure sharing of projects within large companies

Docs format; otherwise client can only view MS Office documents. Currently, Google Drive is integrated with the following features: events tracking, selective folder syncing, version history, simultaneous document editing, commenting on files, sharing permission settings, and an online document editor. For maintaining collaboration with other Google Drive account holders, each client must hold a Google Drive account. Whenever collaborators edit or update a document, it will be synced to Google Drive. An update notification will be received by the registered user. Like Dropbox users, Google Drive account holders can share files by sending a link.

C. ONE DRIVE

One Drive, a storage option created by Microsoft, has iOS and Android OS of mobile versions and Windows and Mac OS desktop versions; however, unlike Dropbox and the same as Google Drive, it has no native support for Linux and relies on third party programs. One Drive is the only service that has an app for Windows Phone and Xbox. One Drive offers 15 GB of free storage space for each registered user, while offering the cheapest upgrade plans among the others in this study. The maximum file upload size is restricted to 10 GB per user.

Integrating Microsoft Web App features, One Drive users can view and edit MS Office file types. The website versions of One Drive supports unique file types such as .WMV, .MP4, PowerPoint, and image. Currently, One Drive is integrated with the following features: version history, events tracking, commenting on files, simultaneous document editing, sharing permission settings, MS Office Web Apps (Word, PowerPoint, Excel, and OneNote), and remote access to files.

D. BOX

Box has applications for iOS, Windows Phone, Android, and BlackBerry mobile phones and Windows and Mac OS desktop versions. Box offers 10 GB of free storage space for each registered user. However, they provide a costly upgrade plan compared with the other PCS services. The maximum file upload size is restricted to 5 GB for the paid personal plan.

Box provides a privacy control over clients' documents. Registered Box users can set permissions for other users in terms of the view, access, edit, and upload operations. In a business, authenticity is a significant issue which Box can provide to its users through password protection. Useful apps such as NetSuite and Salesforce are used to save users' documents to Box. Box allows users to open, edit, and update files with the help of plug-ins, such as Adobe, MS Office. For maintaining collaboration with colleagues, registered Box users can invite all types of users to access shared documents. Box is popular in enterprise and business due to the authenticity and security it offers for shared documents in registered organizations; however, Box is also suitable for personal use.

E. SUMMARY

A simple comparison of the different PCS services highlighted in this paper is presented in Table 1. As can be seen from this table, each PCS service has several pros and cons in terms of operating system compatibility, file size, performance, and capabilities.

IV. QUALITATIVE ANALYSIS

This section describes research on the capabilities of distinct PCS services. Each PCS service implements unique capabilities in order to optimize storage utilization and boost transfer speeds in particular situations [58], [59]. The client capabilities of individual PCS services can be described as chunking, bundling, deduplication, delta-encoding, and data compression.

A. CHUNKING

Processing large files is a big challenge for all PCS services. Files are transmitted either as a single object (no interruption amid the upload), or in chunks (splitting content into data units). Chunking benefits users who have slow network connections because it offers partial submission recovery when upload failure occurs. A summary of each PCS service's chunking capability is shown in Table 2.

Dropbox allows users to upload large files through multiple chunks. If upload is interrupted, Dropbox resumes

TABLE 2. Summary of chunking capabilities of each PCS.

	Chunking if File Size >	Typical Chunk Size	Chunks Expire After	Method	Parameter
Dropbox	150 MB	4 MB	48 hours	1. PUT 2. POST	1. upload_id 2. offset
Google Drive	Users choice (optional)	8 MB	One week	1. PUT 2. POST	1. chunk 2. last-chunk 3. trailer
OneDrive	10 MB	10 MB	Session ends	1. PUT 2. POST	1. uploadUrl 2. nextExpectedRanges
Box	No chunking option	250 MB (segment)	One hour (session ends)	1. POST	1. sessionURL 2. document_id

uploading when possible. Chunks can be any size up to 150 MB. Utilizing a large chunk size can achieve faster throughput as well as fewer calls to the /chunked_upload method. However, once data transmission is interrupted, the upload will have to resume at the beginning of the chunk at which it was interrupted. In this regard, using a smaller chunk size is much more beneficial than a larger one. Calling the PUT method with upload_id as a parameter, infers that uploading to the server is in-progress. The server will initiate a new upload session once the parameter is blank. Another parameter, byte offset of each chunk, denotes the file length from the beginning. Using the POST method, the completed upload is committed by the server.

Google Drive provide a ''resumable upload'' option for reliable transfer; this is especially important with larger files. It consists of three steps: (1) Start a session using the PUT or POST method. (2) Save the session uniform resource identifier. (3) Upload the file using the PUT method. A series of chunks, along with their size, followed by a trailer (optional) are encoded and prepared in order to transfer. In this way, the received message is also verified by transferring dynamic message content. During uploading, several error handling HTTP requests are used to track upload status.

OneDrive has a ''resumable upload'' option for large files, which consists of two steps: (1) Create an upload session using the POST method. (2) Upload fragments. In order to upload a file or a portion of a file, a PUT request to the uploadUrl is received when the session is created. Uploading provides two choices: upload the entire file in one fragment (up to 60 MB) or break the file into multiple fragments and upload each one separately. Fragments must be uploaded in order. An error handling HTTP request is also used here to track upload status. The Box API does not support uploading files in chunks but uses a multipart upload method. However, it is desirable to split larger files when they are larger than 250 MB. The uploading task utilizes the POST method using sessionURL and document_id as parameter settings.

B. BUNDLING

In order to transfer a batch of files while reducing control overhead as well as transmission latency, multiple files are bundled into a single object and pipelined. Dropbox implements and allows bundling of smaller chunks, which increases the amount of data sent per storage operation. Other PCS services simply reuse Transmission Control Protocol (TCP) connections and do not employ any file-bundling strategy.

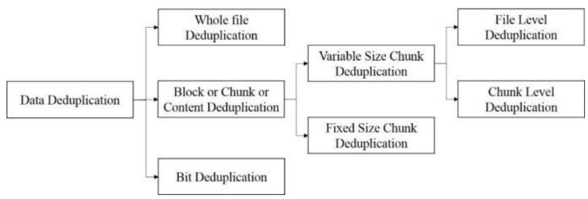

FIGURE 1. Deduplication method [67].

C. DEDUPLICATION

Deduplication is a technique to avoid re-transmitting content that is already available on the servers. In [60]–[62], types of data deduplication are categorized into three level: whole file, block/chunk/content, and bit (Figure 1). Whole file deduplication or single instance storage [63] calculates the file index (hash value) for the whole file. During transmission, duplication may happen if the previously recorded file index is matched with the upcoming file. However, the stored file will only be updated if the upcoming file has a new file index. In this way, a single occurrence of the file is saved and consequent copies are replaced with a pointer to the first record. Depending on the file size, in block deduplication [62], [64], blocks are further divided into fixed and variable size chunks. Utilizing a hash algorithm (SHA-1 [65] or MD5 [66]) a unique ID is created for both fixed and variable size chunks. Variable size chunks are then further divided into file level and chunk level for deduplication. The most efficient method looks at individual blocks and bits within a file and only unique iterations are saved.

During data transfer, the copies of user data are firstly identified in order to conserve upload capacity. This can be accomplished by calculating a hash value using the file content (e.g., SHA256 is used by Dropbox [47]). The hash value is sent to servers prior to submitting the complete file. Uploading repeated files will be skipped when the server determines that the hash value is already stored in the system. Dropbox implements a deduplication technique for

TABLE 3. Summary of the capabilities implemented in each PCS service [2].

	Chunking	Bundling	Deduplication	Delta-encoding	Data Compression
Dropbox	Yes	Yes	Yes	Yes	Always
Google Drive	Yes	No	No	No	Smart
OneDrive	Yes	No	No	No	Never
Box	No	No	No	No	Never

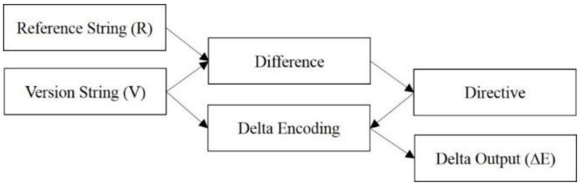

FIGURE 2. Data transmission using delta compression.

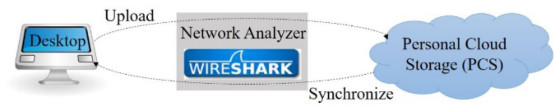

FIGURE 3. Testbed setup for analysis of each PCS service.

TABLE 4. PCS service versions used in this study.

PCS Service	Version
Dropbox	3.8.6.0
Google Drive	1.24.9931.5480
One Drive	17.3.5907.716
Box	4.0.6567.0

identifying replicas of clients' documents, even after they are deleted and later restored. The other PCS services do not implement deduplication.

D. DELTA-ENCODING

Delta-encoding is a technique for storing or transmitting only modified data between revisions by calculating the differences among them. Figure 2 shows the data transmission technique using delta compression. Two strings–a reference string R and a version string V–are assumed to have similar content. Delta-encoding transmits the constructed bit sequence ΔE, which contains the difference between R and V as compactly as possible.

Indeed, delta-encoding provides similar benefits as the combination of chunking and deduplication but with a finer granularity. It may have a positive impact on performance when files are frequently changed (e.g., when people perform collaborative/iterative work). On the other hand, the storage of static content is not affected by this feature. Dropbox has already implemented a delta-encoding technique for uploading modified parts of user data rather than all the data, while other PCS services are attempting to implement it.

E. DATA COMPRESSION

During transmission, the technique of data compression [68] is used to reduce the number of bits, which in turn reduces overall traffic and storage requirements. Dropbox and Google Drive compress data before transmission, which results in less network traffic when compared to the benchmark size, while other PCS services have not yet implemented this technique.

F. SUMMARY

Table 3 shows the capabilities of individual PCS services focusing on: (1) whether each service implements bundling, deduplication, and delta-encoding, (2) the used threshold to split files in chunks (or whether the threshold is variable), and (3) whether content is compressed always, never, or based on file formats (smartly).

V. EMPIRICAL PERFORMANCE ANALYSIS OF PCS SERVICE

After documenting the architecture and capabilities of each service, we now quantify the impact of design choices on performance with diverse benchmark data size. Firstly, we focus on testbed setup for measuring the performance of each PCS service through traffic analysis, transmission rate analysis, synchronization time, completion time, and a protocol overhead calculation.

A. TESTBED SETUP

This research is built around the testbed outlined in Figure 3. As can be seen in the figure, Wireshark [69]–[76], which is a network analysis tool, is used to analyze each packet of internet traffic. From upload to synchronization, each packet is analyzed with specific measurements such as packet number, time, source address (IP/MAC), destination address (IP/MAC), protocol used, length in bytes, and information (packet type details).

Table 4 shows the version of each PCS service that was installed on the desktop machine we used for network analysis. Table 5 shows the setup of the benchmark data sets used. Later, this data sets were used to measure performance and

TABLE 5. Benchmark data sets used to evaluate PCS service performance.

Set	#1	#2	#3	#4	#5	#6	#7
Type	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG
#Files	1	2	4	13	26	130	265
Data Size	1 MB	5 MB	10 MB	50 MB	100 MB	500 MB	1GB

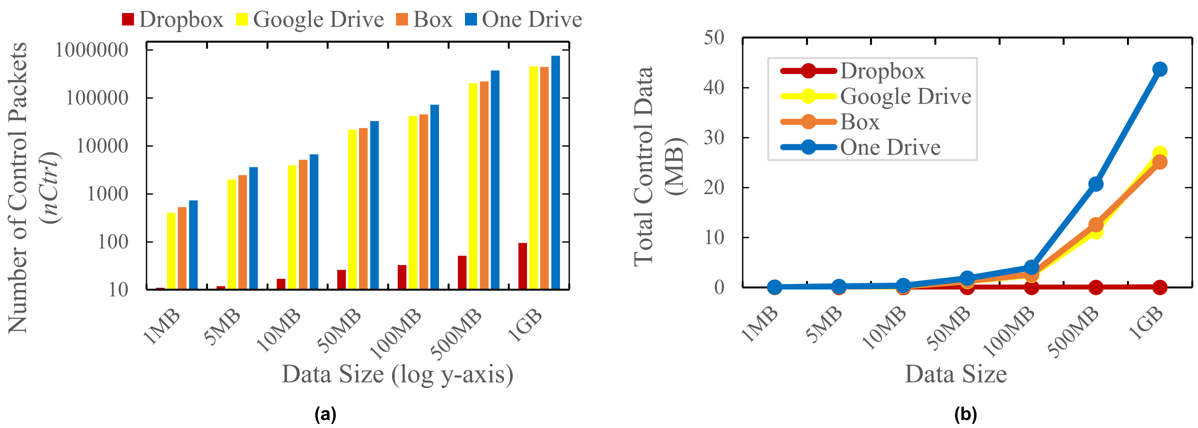

FIGURE 4. Analysis of number and size of control data packets for each PCS service. (a) Number of control data packets. (b) Size of control data packets.

capabilities of each PCS service. This study used only on JPEG images for traffic monitoring.

B. TRAFFIC ANALYSIS

In this section, the complete traffic for each PCS service is analyzed through a diverse data set. In this regard, we subdivide the whole traffic into 4 major parts: control data packet, application data packet, other data packet, and protocol analysis.

1) CONTROL DATA PACKET

The criteria for calculating total control data packets is to sum up the packets related to the control data followed by only TCP protocol during transmission according to the following equation:

$$nCtrlT = |Hnd| + |ACK| + |AUnseen| + |Dup| + |WinUpdate| \quad (1)$$

where $nCtrlT$ denotes the total number of control packets, Hnd stands for three-way handshaking message packet, ACK stands for acknowledgment packet, $AUnseen$ stands for TCP ACKed unseen segment packet, Dup stands for TCP Dup ACK packet, and $WinUpdate$ stands for TCP window update.

During transmission, Hnd occurs when a three-way ([SYN], [SYN, ACK], [ACK]) handshaking message passing is done between sender and receiver. ACK occurs to denote the acknowledgment of any TCP packet. $AUnseen$ occurs when capture is not able to record all packets coming in due to low configuration of machine. A clear indication of a dropped/missing packet (Dup) is occurs when the same ACK number is seen and it is lower than the last byte of data from the sender. A window update occurs when an ACK packet does not acknowledge any more additional data, but only expands the window. Figure 4 shows the total number of control data packets according to (1) and the size of control data (in MB) for each PCS service over different data transmission sizes. It is noted that Google Drive and Box utilized a similar number of control packets (4.5×10^5 for 1 GB), while One Drive uses more (7.6×10^5 for 1 GB). All services exhibit a high volume of traffic compared to Dropbox, which requires relatively few control packets (95 for 1 GB data) to complete transmission.

In the case of the amount of control data packets used, it can be seen that for 1 GB of data being uploaded the of 0.005 MB used by Dropbox, is very low compared with Google Drive, One Drive, and Box, which used between 25 and 43 MB for a 1 GB data upload. As a result, the average percentages of control data that made up the total transmitted data were 3%, 2%, 2%, and 3% for Dropbox, Google Drive, Box, and One Drive, respectively.

2) APPLICATION DATA PACKET

The criteria for calculating total application data packets is to sum up the packets related to the application data followed by TCP, TLSv1, and TLSv1.2 protocols during transmission according to the following equation:

$$nAppT = |AppData| + |Cont| + |PDU| + |OOO| + |PSNC| + |RTrans| + |WinF| \quad (2)$$

where $nAppT$ denotes the total number of application data packets, $Cont$ stands for continuation data, $AppData$ stands for application data, PDU stands for TCP segment of a

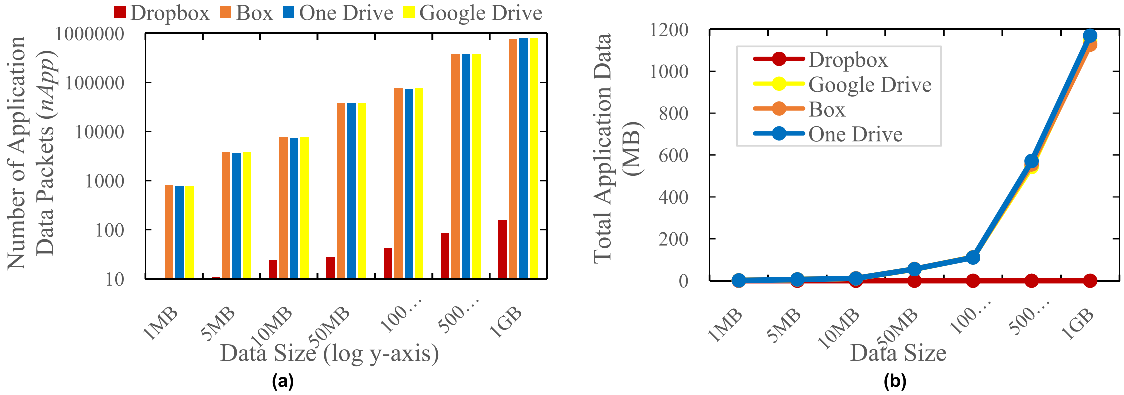

FIGURE 5. **Analysis of number and size of application data packets for each PCS service. (a) Number of application data packets. (b) Size of application data packets.**

reassembled PDU packet, *OOO* stands for TCP out-of-order, *PSNC* stands for TCP previous segment not captured, *RTrans* stands for TCP retransmission (fast and spurious), and *WinF* stands for TCP window full.

During transmission, *AppData* is the actual application data packets, *Cont* is the continuation of data packets, and *PDU* occurs by reassembly of PDUs spanning multiple TCP segments, when the packet contains part of a packet for a protocol that runs on top of TCP. *OOO* occurs when an application data packet is seen with a sequence number lower than the previously received packet on the same connection. *PSNC* occurs when a packet arrives with a sequence number greater than the next expected sequence number on the same connection. *RTrans* occurs when the sender retransmits (general, fast, and spurious) a packet before the expiration of the acknowledgment timer. *WinF* occurs to stop data sending when the payload data in a segment completely fills the Rx buffer on the host on the other end of the TCP session.

Figure 5 shows the total number of application data packets according to (2) and the size of application data (in MB) over different data transmission sizes for each PCS service. It is noted that Google Drive, One Drive, and Box utilized a similar number of application data packets (8×10^5 for 1 GB). All services exhibited a high volume of traffic compared to Dropbox, which requires few application data packets (153 for 1 GB data) to complete transmission. In the case of size of control data packets, it can be seen that for a 1GB data upload, Dropbox utilized 0.2 MB, which is relatively very low compared to One Drive, Google Drive, and Box, which utilized between 1127 and 1170 MB for a 1 GB data transmission. In this regard, on average, application data accounted for 81%, 98%, 98%, and 96% of the total transmitted data for Dropbox, Google Drive, Box, and One Drive, respectively.

3) OTHER PACKETS

The criteria for calculating total other packets is to sum up all the packets excluding control data packets and application data packets followed by TLSv1, TLSv1.2, and SSLv2 protocols during transmission according to the following equation:

$$nOtherT = |Enc| + |Hello| + |EHnd| + |IgnUR| \qquad (3)$$

where *nOtherT* denotes the total number of other packets, *Enc* stands for encrypted data, *Hello* stands for hello packets, *EHnd* stands for encrypted handshake message, and *IgnUR* stands for ignored unknown records. During transmission *Enc* occurs to encrypt the conversation between sender and receiver. *Hello* occurs to enable communication between server and client. *EHnd* occurs to encrypt session traffic when encrypted handshake contains the session key. *IngUR* occurs when TLS record type or version is not able to be detected from the packet.

Figure 6 shows the total number of other packets according to (3) and the size of other data (in MB) for each PCS service over different data transmission sizes. It is noted that One Drive utilized the highest number of other packets (10299 for 1 GB). One Drive and Box utilized 1162 and 295 packets, respectively, for a 1 GB transmission. All services exhibited a higher volume of traffic than Dropbox, which requires relatively few other packets (13 for 1 GB data) to complete transmission. In the case of the size of control data packets, it can be seen that for 1 GB of data being uploaded, Dropbox utilized 0.008 MB, whereas Google Drive, Box, and OneDrive utilized 0.3, 1.7, and 14.8 MB, respectively, for 1 GB of transmitted data. In this regard, other data accounted for 16% of the total transmitted data on average for Dropbox, while this was almost zero percent for Google Drive, Box, and One Drive.

4) PROTOCOL ANALYSIS

During data transmission, the first packet of any traffic is a domain name system (DNS) query, which is responsible for finding a name-specific IP address. Examples of Dropbox names are client-cf.dropbox.com and client-lb.dropbox.com). Examples of Box names are api.box.com, upload.box.com, and client-log.box.com. An example of a Google Drive name is clients3.google.com, and an example of a One Drive name

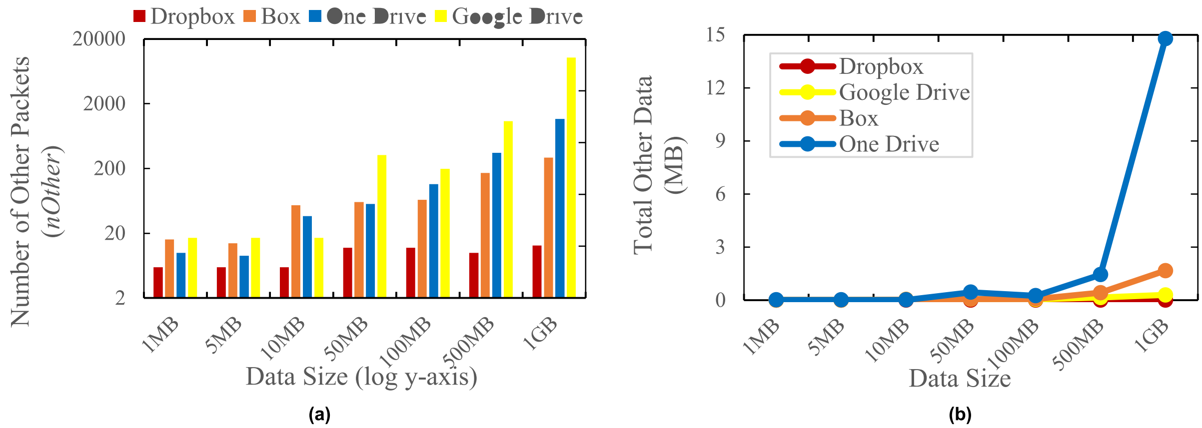

FIGURE 6. Analysis of number and size of other packets for each PCS service. (a) Number of other packets. (b) Size of other packets.

TABLE 6. Number of hello packets/three-way handshakes for different data transmission sizes.

	Dropbox	One Drive	Box	Google Drive
1MB	2/1	2/1	4/2	6/3
5MB	2/1	6/3	4/2	6/3
10MB	2/1	8/4	18/9	6/3
50MB	4/2	14/7	20/10	12/6
100MB	4/2	20/10	22/11	18/9
500MB	4/2	32/16	34/17	70/35
1GB	4/2	32/16	42/21	136/68

TABLE 7. Protocol variations for each PCS service for different data transmission sizes.

Data Transmission Size	Dropbox	Google Drive	Box	One Drive
1 MB	DNS, TCP, TLSv1	DNS, TCP, TLSv1	DNS, TCP, TLSv1.2	DNS, TCP, TLSv1.2
5 MB	DNS, TCP, TLSv1	DNS, TCP, TLSv1	DNS, TCP, TLSv1.2	DNS, TCP, TLSv1.2
10 MB	DNS, TCP, TLSv1	DNS, TCP, TLSv1	DNS, TCP, TLSv1.2	DNS, TCP, TLSv1.2, SSLv2
50 MB	DNS, TCP, TLSv1	DNS, TCP, TLSv1	DNS, TCP, TLSv1.2, SSLv2	DNS, TCP, TLSv1.2, SSLv2
100 MB	DNS, TCP, TLSv1	DNS, TCP, TLSv1	DNS, TCP, TLSv1.2, SSLv2	DNS, TCP, TLSv1.2, SSLv2
500 MB	DNS, TCP, TLSv1	DNS, TCP, TLSv1	DNS, TCP, TLSv1.2, SSLv2	DNS, TCP, TLSv1.2, SSLv2
1 GB	DNS, TCP, TLSv1	DNS, TCP, TLSv1	DNS, TCP, TLSv1.2, SSLv2	DNS, TCP, TLSv1.2, SSLv2

is dm2304.storage.live.com. DNS protocol is used to take queries and responses with a specific IP address. TCP is used to continue transmission by sending control data packets. A three-way handshaking message is also exchanged through TCP before passing on application data packets. Table 6 shows the number of hello packets (from client and server) and the number of three-way handshaking messages conducted for each PCS service for different data sizes.

Finally, application data can be passed through either secret channel, secret data or both using transport layer security (TLS) protocol. The versions selection (v1 or v1.2) of TLS varies between PCS services. For encrypted data transmission, secure sockets layer (SSLv2) protocol is used to encrypt data using a secret key. Table 7 shows the different protocols that are used by each PCS service based on data size.

C. IMPACT OF DATA SIZE ON NUMBER OF PACKETS AND TRANSMISSION RATE

During transmission the total number of packets and average data transmission speed for each PCS service with different data transmission sizes can be calculated according to the following equations:

$$nPac = |nCtrlT| + |nAppT| + |nOtherT| \quad (4)$$

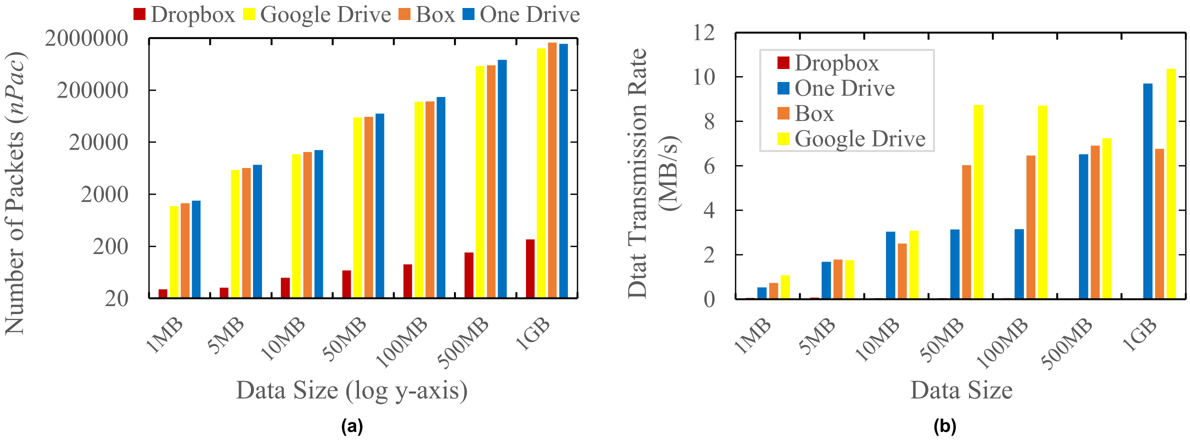

FIGURE 7. Total number of packets and transmission rate for different data transmission sizes. (a) Total number of packets. (b) Average data transmission rate.

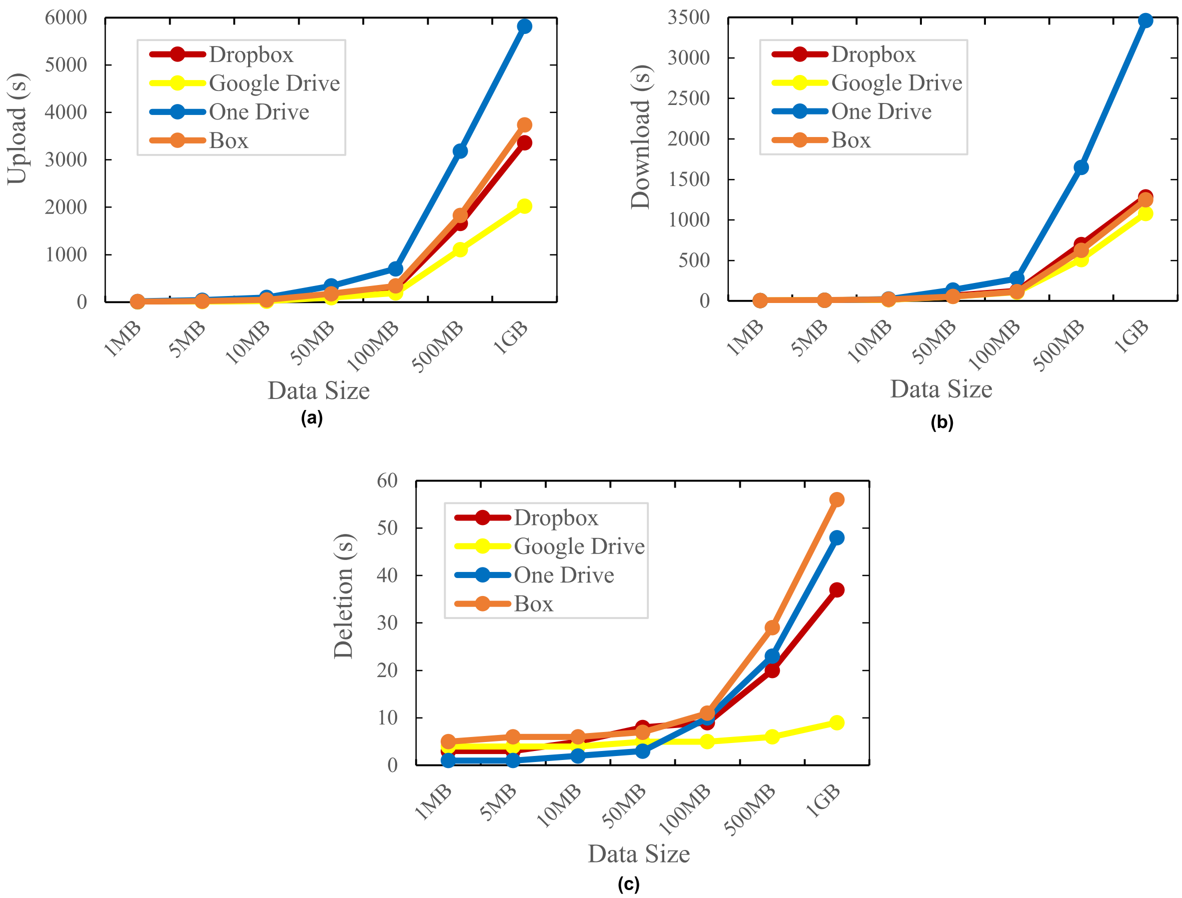

FIGURE 8. Duration for upload, download, and deletion operations for each PCS service. (a) Upload duration. (b) Download duration. (c) Deletion duration.

$$aSpeed = \frac{\sum_{i=1}^{n} PacL_i \times 8/10^6}{tComp} \quad (5)$$

where *nPac* denotes total number of packets, which is summation of control, application, and other data packets; *aSpeed* stands for average data transmission speed (MB/s); $PacL_i$ stands for i^{th} packet's length (Bytes), $i = 1, 2, \ldots\ldots .n$; n denotes the number of packets; and *tComp* stands for completion time (s).

Figure 7 shows the number of packets and data transmission rate over data transmission size. It is noted that Google Drive, One Drive, and Box exhibit similar number of total packets ($>10^6$ for 1 GB). All services exhibit a higher volume of traffic than Dropbox, which requires relatively few packets (270 for 1 GB data) to complete transmission. In the case of transmission rate, it can be seen that Dropbox exhibits similar average speed for all data transmission sizes. Google Drive, One Drive, and Box's transmission rates increase initially and then show little fluctuation for higher data transmission sizes.

D. OVERALL PERFORMANCE

Figure 8 shows the overall duration of upload, download, and deletion operations (in seconds) of distinct PCS services for different data transmission sizes. All operations are performed on PCS services' website including file syncing. It can be seen that One Drive consumed the largest amount of time for uploads (5818s for 1 GB), while Box and Dropbox had upload times of 3740s and 3360s, respectively. Google Drive had the fastest upload speed (2026s for 1 GB). For downloading, One Drive was the slowest (3461s for 1 GB), while Dropbox, Google Drive, and Box all took between 1080s and 1286s for a 1 GB download. Figure 8(c) shows the deletion duration (in seconds), which increased gradually with data size for all PCS services.

E. SYNCHRONIZATION INITIALIZATION TIME

The criteria for calculating the synchronization time is to sum up the time required for each request and reply according to the following equation:

$$tSync = \sum\nolimits_{i=1}^{j} tDNS_i + \sum\nolimits_{i=1}^{k} tHnd_i \qquad (6)$$

where $tSync$ denotes total synchronization time, $tDNS_i$ stands for the time between the i^{th} DNS query and answer, and $tHnd_i$ stands for time taken to complete the i^{th} three-way ([SYN], [SYN, ACK], and [ACK]) handshaking message.

Figure 9 shows the increasing tendency of $tSync$ for each PCS service over different data sizes. Dropbox exhibited the fastest storage with lowest $tSync$ (9.3s), while the slowest storage was One Drive (67.4s) for a 1 GB data uploaded. However, Google Drive still has a low synchronization time (10.4s), while Box had a relatively high synchronization time (52.4s) for a 1 GB upload.

F. COMPLETION TIME

The completion time is calculated as the time difference between the first the packet and the last packet in any storage flow according to the following equation:

$$tComp = \sum_{i=1}^{n} PacT_i \qquad (7)$$

where $tComp$ denotes the completion time, and $PacT_i$ stands for time since the i-1^{th} packet.

Figure 10 shows how Dropbox exhibits faster completion time than the other PCS services. Google Drive and One Drive achieved similar performance, and Box was the slowest. For example, to upload 1 GB data, the completion time

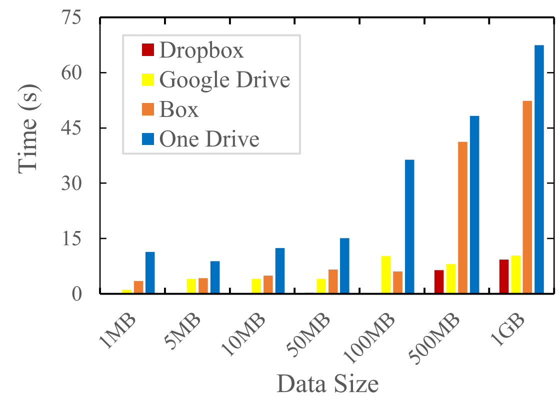

FIGURE 9. Synchronization time for each PCS service for different data sizes.

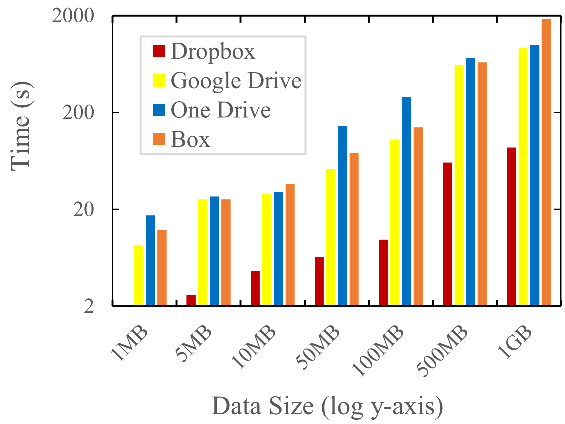

FIGURE 10. Completion time for each PCS service for different data sizes.

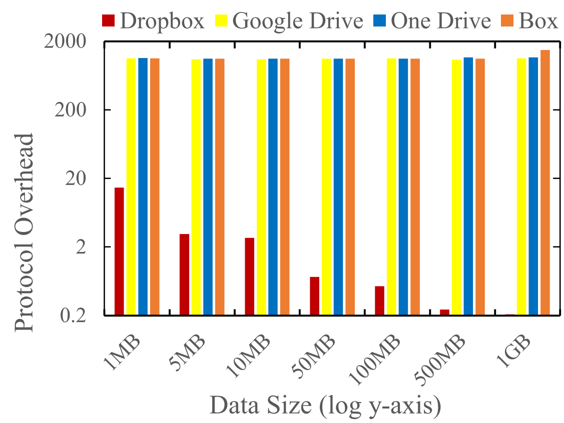

FIGURE 11. Protocol overhead for each PCS service for different data sizes.

achieved for Dropbox, Google Drive, One Drive and Box was 87s, 903s, 967s, and 1808s, respectively.

G. PROTOCOL OVERHEAD

The criteria for calculating the protocol overhead is to calculate the ratio of total storage and control traffic to the

TABLE 8. Summary of the performances for each PCS service over various data sizes.

	Dropbox	Google Drive	One Drive	Box
Control Data Packet for Different Sizes	Lowest	Highest	Average	Average
Application Data Packet for Different Sizes	Lowest	Average	Average	Average
Other Data Packet for Different Sizes	Lowest	Average	Highest	Average
Total number of packets and Transmission Rate	Lowest	Average	Average	Average
Upload Duration	Fastest	Average	Average	Slowest
Download Duration	Average	Average	Slowest	Average
Deletion Duration	Average	Fastest	Average	Slowest
Hello Packet and Three-Way Handshaking	Lowest	Highest	Average	Average
Synchronization Time	Fastest	Average	Average	Slowest
Completion Time	Fastest	Average	Slowest	Average
Protocol Overhead	Lowest	Average	Average	Average

benchmark size according to the following equation [24]:

$$pOver_j = \frac{\sum_{i=1}^{n} PacL_i}{bSize_j} \quad (8)$$

where $pOver_j$ denotes protocol overhead for the j^{th} data size, $bSize_j$ stands for the benchmark size for the j^{th} data size, and $j = 1, 2, \ldots, 7$. Figure 11 depicts the protocol overhead for individual PCS services. It can be seen that the overhead for Google Drive, One Drive, and Box was almost the same for all data sizes. However, it is clear that Dropbox exhibits a decreasing tendency as data size increases due to its capabilities, such as bundling, deduplication, and compression. The lack of bundling dramatically increases overhead when multiple small files are uploaded, because the upload of every file requires application layer control traffic to be sent.

H. SUMMARY

Table 8 summarizes our performance evaluation for different factors, such as the average performance for each packet type (application, control, and other) over a range of sizes, operation (upload, download, and deletion) duration, synchronization time, completion time, and protocol overhead for each PCS service. Considering the diverse benchmark size, application data accounted for the highest proportion of all transmitted data, as compared with control and other data during testbed analysis.

It can be seen that Dropbox utilizes its above mentioned capabilities well in order to enhance performance. Google Drive, SkyDrive, and Box are all inferior to Dropbox because they only implement some capabilities.

VI. CONCLUSION

In this paper, we analyzed the potentials and limitations of different PCS services for managing mobile multimedia PHRs by conducting a qualitative and quantitative analysis. The qualitative analysis investigated PCS capabilities, including chunking, bundling, deduplication, delta-encoding, and data compression. We performed quantitative analysis using DropBox, Google Drive, One Drive, and Box, which allowed us to model traffic analysis, transmission rate analysis, synchronization time, completion time, and protocol overhead evaluation for diverse data sets. Testbed results demonstrated that average transmitted data of 94%, 3%, and 4% was achieved for application data, control data, and other data, respectively. Besides, testbed results on Dropbox show that it outperforms the other PCS services considered in this study. The superior performance of Dropbox is predominantly due to its combining of PCS capabilities. Our future objective is to develop a user-friendly, reliable, and fast PCS service, which includes all PCS capabilities presented in this paper in order to maximize PCS performance.

REFERENCES

[1] X. Liu, Y. Xia, W. Yang, and F. Yang, "Secure and efficient querying over personal health records in cloud computing," *Neurocomputing*, vol. 274, pp. 99–105, Jan. 2018.
[2] A. M. C. de Araújo, V. C. Times, and M. U. da Silva, "A cloud service for graphical user interfaces generation and electronic health record storage," in *Information Technology-New Generations*. Cham, Switzerland: Springer, 2018, pp. 257–263.
[3] M. A. Alyami, M. Almotairi, L. Aikins, A. R. Yataco, and Y. T. Song, "Managing personal health records using meta-data and cloud storage," in *Proc. IEEE/ACIS 16th Int. Conf. Comput. Inf. Sci.*, May 2017, pp. 265–271.
[4] K. Belyaev, W. Sun, I. Ray, and I. Ray, "On the design and analysis of protocols for personal health record storage on personal data server devices," *Future Gener. Comput. Syst.*, vol. 80, pp. 467–482, Mar. 2018.
[5] G. D. Gonçalves, I. Drago, A. V. Borges, A. P. Couto, and J. M. de Almeida, "Analysing costs and benefits of content sharing in cloud storage," in *Proc. Workshop Fostering Latin-Amer. Res. Data Commun. Netw.*, 2016, pp. 43–45.
[6] R. Gracia-Tinedo *et al.*, "Giving wings to your data: A first experience of personal cloud interoperability," *Future Gener. Comput. Syst.*, vol. 78, pp. 1055–1070, Jan. 2018.
[7] K. Umarani and M. Khanna, "Privacy preserving for remote data based on identity with high performance for cloud storage," *Int. Res. J. Eng. Technol.*, vol. 4, no. 5, pp. 1062–1066, 2017.
[8] J. Wang, "Critical factors for personal cloud storage adoption in China," *J. Data Inf. Sci.*, vol. 1, no. 2, pp. 60–74, 2016.
[9] R. Singh, A. Abidi, and M. A. Qadeer, "SyncWorld: A cloud storage/synchronization service using java and php," in *Proc. Wireless Opt. Commun. Netw.*, Jul. 2016, pp. 1–5.
[10] K. Wu, J. Vassileva, and Y. Zhao, "Understanding users' intention to switch personal cloud storage services: Evidence from the Chinese market," *Comput. Hum. Behav.*, vol. 68, pp. 300–314, Mar. 2017.
[11] B. Hou, F. Chen, Z. Ou, R. Wang, and M. Mesnier, "Understanding I/O performance behaviors of cloud storage from a client's perspective" *ACM Trans. Storage*, vol. 13, no. 2, p. 16, 2017.
[12] P. S. Jagtap and A. D. Gujar, "Survey on cloud backup services of personal storage," *Int. J.*, vol. 4, no. 1, pp. 1–4, 2016.
[13] P. Windley. *From Personal Computers to Personal Clouds*. Accessed: Apr. 23, 2012. [Online]. Available: http://www.windley.com/archives/2012/04/from_personal_computers_to_personal_clouds.shtml
[14] F. E. Gillett *et al.*, The personal cloud: transforming personal computing, mobile, and Web markets," CiteULike, Madison, WI, USA, Tec. Rep. 9539902, Jun. 2011.

[15] Gartner. *Cloud Computing*. Accessed: Oct. 5, 2015. [Online]. Available: http://www.gartner.com/technology/topics/cloud-computing.jsp#

[16] D. Dai, W. Zheng, and T. Fan, "Evaluation of personal cloud storage products in China," *Ind. Manage. Data Syst.*, vol. 117, no. 1, pp. 131–148, 2017.

[17] R. I. Heinsen, C. P. Lopez, T. D. T. Nguyen, and E.-N. Huh, "Cloud storage federation as a service reference architecture," in *Proc. Int. Conf. Inf. Sci. Appl.* Singapore, Springer, 2017, pp. 668–675.

[18] M. Hajjat *et al.*, "Cloudward bound: Planning for beneficial migration of enterprise applications to the cloud," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 243–254, 2011.

[19] R. Gracia-Tinedo, M. S. Artigas, A. Moreno-Martinez, C. Cotes, and P. G. Lopez, "Actively measuring personal cloud storage," in *Proc. 6th IEEE Conf. Cloud Comput.*, Jun. 2013, pp. 301–308.

[20] W. Hu, T. Yang, and N. J. Matthews, "The good, the bad and the ugly of consumer cloud storage," *ACM SIGOPS Oper. Syst. Rev.*, vol. 44, no. 3, pp. 110–115, 2010.

[21] Z. Li *et al.*, "Towards network-level efficiency for cloud storage services," in *Proc. ACM Conf. Internet Meas.*, 2014, pp. 115–128.

[22] A. Lenk, M. Klems, J. Nimis, S. Tai, and T. Sandholm, "What's inside the cloud? An architectural map of the cloud landscape," in *Proc. ICSE Workshop Softw. Eng. Challenges Cloud Comput.*, May 2009, pp. 23–31.

[23] A. Iosup, N. Yigitbasi, and D. Epema, "On the performance variability of production cloud services," in *Proc. 11th IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput. (CCGrid)*, May 2011, pp. 104–113.

[24] A. Li, X. Yang, S. Kandula, and M. Zhang, "Comparing public-cloud providers," *IEEE Internet Comput.*, vol. 15, no. 2, pp. 50–53, Mar. 2011.

[25] G. Wang and T. S. E. Ng, "The impact of virtualization on network performance of amazon ec2 data center," in *Proc. INFOCOM*, Mar. 2010, pp. 1–9.

[26] E. Bocchi, I. Drago, and M. Mellia, "Personal cloud storage benchmarks and comparison," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 751–764, Oct. 2017.

[27] I. Drago, M. Mellia, M. M. Munafo, A. Sperotto, R. Sadre, and A. Pras, "Inside dropbox: understanding personal cloud storage services," in *Proc. ACM Conf. Internet Meas. (IMC)*, 2012, pp. 481–494.

[28] E. Walker, "Benchmarking Amazon EC2 for high-performance scientific computing," *LOGIN*, vol. 33, no. 5, pp. 18–23, Oct. 2008.

[29] C. Schwartz, M. Hirth, T. Hoßfeld, and P. Tran-Gia, "Performance model for waiting times in cloud file synchronization services," in *Proc. 26th Int. Conf. Teletraffic Congr. (ITC)*, Sep. 2014, pp. 1–9.

[30] P. Casas, H. R. Fischer, S. Suette, and R. Schatz, "A first look at quality of experience in personal cloud storage services," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Jun. 2013, pp. 733–737.

[31] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, "The nature of data center traffic: Measurements & analysis," in *Proc. 9th ACM SIGCOMM Conf. Internet Meas.*, 2009, pp. 202–208.

[32] R. Clarke, "How reliable is cloudsourcing? A review of articles in the technical media 2005–11," *Comput. Law Secur. Rev.*, vol. 28, no. 1, pp. 90–95, 2012.

[33] M. Anti. *Behind the Great Firewall of China*. Accessed: Jul. 30, 2012. [Online]. Available: https://www.youtube.com/watch?v=yrcaHGqTqHk

[34] G. Greenwald and E. MacAskill, "NSA Prism program taps into user data of Apple, Google and others," *Guardian*, vol. 7, no. 6, pp. 1–43, 2013.

[35] B. P. Rimal, E. Choi, and I. Lumb, "A taxonomy and survey of cloud computing systems," in *Proc. 5th Int. Joint Conf. INC, IMS IDC (NCM)*, 2009, pp. 44–51.

[36] M. Armbrust *et al.*, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[37] D. Durkee, "Why cloud computing will never be free," *Commun. ACM*, vol. 53, no. 5, pp. 62–69, 2010.

[38] C. N. Höfer and G. Karagiannis, "Cloud computing services: Taxonomy and comparison," *J. Internet Services Appl.*, vol. 2, no. 2, pp. 81–94, 2011.

[39] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: Towards a cloud definition," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 50–55, 2008.

[40] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: State-of-the-art and research challenges," *J. Internet Services Appl.*, vol. 1, no. 1, pp. 7–18, May 2010.

[41] E. Hamburger. 2012. Google Drive vs. Dropbox, SkyDrive, SugarSync, and Others: A Cloud Sync Storage Face-off. The Verge. Accessed: Sep. 14, 2018. [Online]. Available: https://www.theverge.com/2012/4/24/2954960/google-drive-dropbox-skydrive-sugarsync-cloud-storage-competition

[42] Chad Brooks. *Cloud Storage Often Results in Data Loss*. Accessed: Oct. 8, 2011. [Online]. Available: http://www.businessnewsdaily.com/1543-cloud-data-storage-problems.html

[43] J. Čurn, *How a Bug in Dropbox Permanently Deleted my 8000 Photos*. Accessed: Jul. 29, 2014. [Online]. Available: https://medium.com/@jan.curn/how-bug-in-dropbox-permanently-deleted-my-8000-photos-cb7dcf13647b.

[44] G. Huntley. *Dropbox Confirms that a Bug Within Selective Sync may have Caused Data Loss*. Accessed: Oct. 11, 2014. [Online]. Available: https://news.ycombinator.com/item?id=8440985

[45] E. Mill. *Dropbox Bug Can Permanently Lose Your Files*. Accessed: Oct. 26, 2012. [Online]. Available: https://konklone.com/post/dropbox-bug-can-permanently-lose-your-files.

[46] J. Cook. *All The Different Ways That 'iCloud' Naked Celebrity Photo Leak Might Have Happened*. Accessed: Sep. 1, 2014. [Online]. Available: http://www.businessinsider.my/icloud-naked-celebrity-photo-leak-2014-9/

[47] M. Mulazzani, S. Schrittwieser, M. Leithner, M. Huber, and E. R. Weippl, "Dark clouds on the horizon: Using cloud storage as attack vector and online slack space," in *Proc. USENIX Secur. Symp.*, 2011, pp. 65–76.

[48] S. Byrne. *Microsoft OneDrive for Business Modifies Files as it Syncs*. Accessed: Apr. 17, 2014. [Online]. Available: http://www.myce.com/news/microsoft-onedrive-for-business-modifies-files-as-it-syncs-71168/

[49] Z. Whittaker. *Dropbox under fire for 'DMCA Takedown' of Personal Folders, but Fears are Vastly Overblown*. Accessed: Mar. 30, 2014. [Online]. Available: http://www.zdnet.com/article/dropbox-under-fire-for-dmca-takedown-of-personal-folders-but-fears-are-vastly-overblown/.

[50] J. Silber. *Shutting Down Ubuntu One File Services*. Accessed: Apr. 2, 2014. [Online]. Available: http://blog.canonical.com/2014/04/02/shutting-down-ubuntu-one-file-services/

[51] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in *Proc. 18th ACM Conf. Comput. Commun. Secur.*, 2011, pp. 491–500.

[52] D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Side channels in cloud services: Deduplication in cloud storage," *IEEE Security Privacy*, vol. 8, no. 6, pp. 40–47, Nov./Dec. 2010.

[53] A. Tridgell and P. Mackerras, "The rsync algorithm," Dept. Comput. Sci., Austral. Nat. Univ., Canberra, ACT, Australia, Tech. Rep. TR-CS-96-05, Jun. 1996.

[54] T. Suel, P. Noel, and D. Trendafilov, "Improved file synchronization techniques for maintaining large replicated collections over slow networks," in *Proc. 20th Int. Conf. Data Eng.*, Mar. 2014, pp. 153–164.

[55] P. G. Lopez, M. Sanchez-Artigas, S. Toda, C. Cotes, and J. Lenton, "StackSync: Bringing elasticity to dropbox-like file synchronization," in *Proc. 15th ACM Conf. Int. Middleware*, 2014, pp. 49–60.

[56] S. Han, H. Shen, T. Kim, A. Krishnamurthy, T. E. Anderson, and D. Wetherall, "MetaSync: File synchronization across multiple untrusted storage services," in *Proc. USENIX Annu. Tech. Conf.*, Santa Clara, CA, USA, 2015, pp. 83–95.

[57] S. Mitroff. *OneDrive, Dropbox, Google Drive and Box: Which Cloud Storage Service is Right for you*. Accessed: Aug. 24, 2015. [Online]. Available: http://www.cnet.com/how-to/onedrive-dropbox-google-drive-and-box-which-cloud-storage-service-is-right-for-you/

[58] A. Muthitacharoen, B. Chen, and D. Mazieres, "A low-bandwidth network file system," *ACM SIGOPS Oper. Syst. Rev.*, vol. 35, no. 5, pp. 174–187, 2001.

[59] A. Tridgell, "Efficient algorithms for sorting and synchronization," Ph.D. dissertation, Dept. Comput. Sci., Austral. Nat. Univ., Canberra, ACT, Australia, 1999, p. 115.

[60] J. Min, D. Yoon, and Y. Won, "Efficient deduplication techniques for modern backup operation," *IEEE Trans. Comput.*, vol. 60, no. 6, pp. 824–840, Jun. 2011.

[61] J. Wei, H. Jiang, K. Zhou, and D. Feng, "MAD2: A scalable high-throughput exact deduplication approach for network backup services," in *Proc. 26th IEEE Symp. Mass Storage Syst. Technol. (MSST)*, May 2010, pp. 1–14.

[62] B. Zhu, K. Li, and R. H. Patterson, "Avoiding the disk bottleneck in the data domain deduplication file system," in *Proc. 6th USENIX Conf. File Storage Technol. (Fast)*, 2008, pp. 1–14.

[63] W. J. Bolosky, S. Corbin, D. Goebel, and J. R. Douceur, "Single instance storage in Windows 2000," in *Proc. 4th USENIX Windows Syst. Symp.*, 2000, pp. 13–24.

[64] D. Bhagwat, K. Eshghi, D. D. E. Long, and M. Lillibridge, "Extreme binning: Scalable, parallel deduplication for chunk-based file backup," in *Proc. IEEE Int. Symp. Modeling, Anal. Simulation Comput. Telecommun. Syst. (MASCOTS)*, Sep. 2009, pp. 1–9.
[65] J. H. Burrows, "Secure hash standard," Dept. Commerce, Washington, DC, USA, Tech. Rep. ADA406543, 1995.
[66] R. Rivest, *The MD5 Message-Digest Algorithm*, document RFC 1321, 1992.
[67] M. S. Devi, V. V. Khanna, and N. A. Bhalaji, "Enhanced dynamic whole file de-duplication (DWFD) for space optimization in private cloud storage backup," *Int. J. Mach. Learn. Comput.*, vol. 4, no. 4, p. 376, 2014.
[68] J. Karppanen, "Lossless differential compression for synchronizing arbitrary single-dimensional strings," M.S. thesis, Dept. Comput. Sci., Univ. Helsinki, Helsinki, Finland, Oct. 2012.
[69] A. Orebaugh, G. Ramirez, and J. Beale, *Wireshark & Ethereal Network Protocol Analyzer Toolkit*. Maryland Heights, MO, USA: Syngress, 2006.
[70] M. Masud, M. S. Hossain, and A. Alamri, "Data interoperability and multimedia content management in e-health systems," *IEEE Trans. Inf. Technol. Biomed.*, vol. 16, no. 6, pp. 1015–1023, Nov. 2012.
[71] Y. Hu, K. Duan, Y. Zhang, M. S. Hossain, S. M. M. Rahman, and A. Alelaiwi, "Simultaneously aided diagnosis model for outpatient departments via healthcare big data analytics," *Multimedia Tools Appl.*, vol. 77, no. 3, pp. 3729–3743, 2018.
[72] A. K. Das *et al.*, "Big media healthcare data processing in cloud: A collaborative resource management perspective," *Cluster Comput.*, vol. 20, no. 2, pp. 1599–1614, 2017.
[73] M. S. Hossain, G. Muhammad, and A. Alamri, "Smart healthcare monitoring: A voice pathology detection paradigm for smart cities," *Multimedia Syst.*, pp. 1–11, Jul. 2017, doi: 10.1007/s00530-017-0561-x.
[74] M. S. Hossain and G. Muhammad, "Healthcare big data voice pathology assessment framework," *IEEE Access*, vol. 4, pp. 7806–7815, 2016.
[75] Y. Hao, L. Peng, H. Lu, M. M. Hassan, and A. Alamri, "Energy harvesting based body area networks for smart health," *Sensors*, vol. 17, no. 7, p. 1602, 2017.
[76] M. S. Hossain, M. Moniruzzaman, G. Muhammad, A. Ghoneim, and A. Alamri, "Big data-driven service composition using parallel clustered particle swarm optimization in mobile environment," *IEEE Trans. Services Comput.*, vol. 9, no. 5, pp. 806–817, Sep. 2016.

MAHMUDA AKTER received the B.Sc. Engg. and M.Sc. Engg. degrees from the Department of Information and Communication Technology, Mawlana Bhashani Science and Technology University, Tangail, Bangladesh, in 2012 and 2015, respectively. From 2012 to 2013, she was also served as an Instructor with the Department of Computer Technology, Royal Institute of Technology, Gazipur, Bangladesh. In 2015, she was served as a Research Assistant with the Department of Computer System and Technology, University of Malaya, Malaysia. She has authored/co-authored three refereed publications. Her major research interests include wireless sensor networks, cloud computing, and their real-world applications.

ABDULLAH GANI received the bachelor's and master's degrees from the University of Hull, U.K., and the Ph.D. degree from The University of Sheffield, U.K. He has vast teaching experience due to having worked in various educational institutions locally and abroad–schools, teaching college, ministry of education, and universities. He is currently an Associate Professor with the Department of Computer System and Technology, Faculty of Computer Science and Information Technology, University of Malaya, Malaysia. His interest in research started in 1983 when he was chosen to attend the scientific research course in RECSAM by the Ministry of Education, Malaysia. He has published over 100 academic papers in conferences and respectable journals. He actively supervises many students at all level of study–bachelor's, master's, and Ph.D. His interest of research includes self-organized systems, reinforcement learning, and wireless-related networks. He is currently involved in mobile cloud computing with high impact research grant of USD 500 000 (RM 1.5M) for the period of 2011–2016.

MD. OBAIDUR RAHMAN (M'12) received the B.Sc. degree in computer science and information technology from the Islamic University of Technology, Dhaka, Bangladesh, in 2003, and the M.S. degree in computer engineering from the Networking Lab and the Ph.D. degree in information and communication from Kyung Hee University, South Korea, in 2008 and 2012, respectively. He is currently a Professor with the Department of Computer Science and Engineering, Dhaka University of Engineering & Technology, Gazipur, Bangladesh. His research interests include medium access control and transport protocol issues in wireless and mobile communication, cognitive networks, and sensor networks. He is a member of the Institute of Engineers, Bangladesh. He has contributed in the wireless communication research by adopting himself as a PC member, a reviewer, and an editor in several international conference and journals.

MOHAMMAD MEHEDI HASSAN (M'11–SM'18) received the Ph.D. degree in computer engineering from Kyung Hee University, South Korea, in 2011. He is currently an Associate Professor with the Information Systems Department, College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Saudi Arabia. He has published over 100 research papers in the journals and conferences of international repute. His research areas of interest are cloud federation, multimedia cloud, sensor-cloud, Internet of Things, big data, mobile cloud, sensor network, publish/subscribe systems, and recommender systems. He received the Best Paper Award from the CloudComp Conference, China, in 2014, and the Excellence in Research Award from CCIS, KSU, in 2015 and 2016, respectively. He has also played a role of a guest editor of several international ISI-indexed journals.

AHMAD ALMOGREN received the Ph.D. degree in computer sciences from Southern Methodist University, Dallas, TX, USA, in 2002. Previously, he was an Assistant Professor of computer science and a member of the Scientific Council, Riyadh College of Technology. He also served as the Dean of the College of Computer and Information Sciences and the Head of the Council of Academic, Al Yamamah University. He is currently an Associate Professor and the Vice Dean for the Development and Quality with the College of Computer and Information Sciences, King Saud University, Saudi Arabia. His research areas of interest include mobile and pervasive computing, computer security, sensor and cognitive networks, and data consistency. He has served as a guest editor for several computer journals.

SHAFIQ AHMAD received the Ph.D. degree from RMIT University, Melbourne, Australia. He has over two decades working experience both in industry and academia in Australia, Europe, and Asia. He is currently an Assistant Professor with the College of Engineering, King Saud University. He has published a research book and over 40 research articles in international journals and conferences. His research interests include smart manufacturing, manufacturing analytics, performance analysis, bibliometric network analysis, and statistical quality control.