Chapter 12
Secure Distributed Storage for the Internet of Things

Sinjoni Mukhopadhyay

12.1 Outline

With recent advances in machine-to-machine communication built on cloud computing and networks of data-gathering sensors, the Internet of Things (IoT) has become an important part of the system and software community. IoT devices generate massive amounts of data which makes the cloud an ideal storage solution for such data. Although cloud storage may be preferred over on-premise storage due to its ease of access, low cost and delegated infrastructure management, IoT data storage using a single cloud provider has several disadvantages. Relying on a single cloud provider to store all the data compromises data reliability while simultaneously sacrificing data availability. IoT devices generate different types of data, all of which are efficiently organized using different storage solutions. This chapter discusses leveraging distributed storage properties to securely store different types of IoT data while simultaneously ensuring data availability.

A few years ago, the idea of placing workloads on a single public or private cloud seemed very enticing; but since the introduction of hybrid cloud architectures [1], the choices in terms of the variety of services available has made it a more attractive option for many enterprises. As more enterprises are avoiding dependency on a single public cloud provider, cloud computing is making a shift towards the multi-cloud strategy. The main difference between multi-cloud and hybrid cloud solutions, in this author's opinion, is the manner in which the resources are deployed in each model. Hybrid cloud solutions can consist of a combination of private and public cloud deployments, all a part of a single cloud service provider, or multiple private or public cloud deployments; whereas multi-cloud solutions can include either private or public cloud deployments belonging to different cloud service pro-

S. Mukhopadhyay (✉)
Computer Science, University of California, Santa Cruz, CA, USA
e-mail: simukhop@ucsc.edu

F. D. Hudson (ed.), *Women Securing the Future with TIPPSS for IoT*, Women in Engineering and Science, https://doi.org/10.1007/978-3-030-15705-0_12

viders. Additionally, all functions in a hybrid cloud are performed as a conjoint effort between public and private clouds, whereas in multi-cloud solutions different services are provided by different cloud vendors. Using a multi-cloud strategy has a certain degree of flexibility which allows the user to choose between a variety of features made available at the most competitive pricing. Organizations also believe that a multi-cloud strategy has other benefits like the following: it helps avoid vendor lock-ins; it tackles cloud reliability; it allows organizations to pick between services that are best suited to their applications/workload, and in addition it also provides benefits of data sovereignty. Data sovereignty defines the idea that information which has been stored in binary/digital form is subjected to the laws of the country in which it is located. The multi-cloud strategy allows organizations to leverage geographically dispersed clouds to meet data sovereignty requirements and improve the user experience.

Distributed cloud storage allows data to be stored across multiple storage servers, either belonging to the same cloud or belonging to different cloud providers. With data being distributed across servers, the following questions arise:

1. How can this distributed data be recovered?
2. How easily can an adversary regroup data chunks to steal the original data?
3. How is data recovered if a subgroup of servers is failing?

This chapter discusses one such distributed storage technique that has major reliability and availability advantages. After introducing the distributed storage technique, we go on to talk about how this technique answers each of these questions.

Secret-sharing is a distributed storage algorithm used to store data chunks across multiple servers. Each server has an erasure coded piece of the data, which reveals no information about the original data. Data can be recovered as long as a sufficient number of servers are functional and available. Additionally, in case of data breaches in a single server, the attacker only gets access to the piece of the data stored under that server, which does not provide sufficient information to generate the original data object. This chapter outlines the existing storage solutions for different types of IoT data. It outlines the limitations of the existing state of the art, while addressing the benefits of applying a secret-sharing based IoT data model.

12.2 Background

Mark Hung, vice-president of Gartner Research stated that "The Internet of Things will have a great impact on the economy by transforming many enterprises into digital businesses and facilitating new business models, improving efficiency and increasing employee and customer engagement [2]." Based on reports provided by Gartner analysts, illustrated in Table 12.1 [3], looking into the future, there will be a 33.7% Compound Annual Growth Rate (CAGR) in units of IoT devices or connected things, from 6.4 billion in 2016 to 20.4 billion in 2020, an overall increase of 220%.

Table 12.1 IoT installed Base by Category (Millions of Units) and CAGR (Compound Annual Growth Rate) [3]

Category	2016	2017	2018	2020	CAGR (%)
Consumer	3963.0	5244.3	7036.3	12,863.0	34.2
Business: Cross-Industry	1102.1	1501.0	2132.6	4381.4	41.2
Business: Vertical-Specific	1316.6	1635.4	2027.7	3171	24.6
Grand Total	*6381.8*	*8380.6*	*11,196.6*	*20,415.4*	*33.7*

Table 12.2 Predicted rise in endpoint spending and CAGR (Millions of Dollars) [3]

Category	2016	2017	2018	2020	CAGR (%)
Consumer	532,515	725,696	985,348	1,494,466	29.4
Business: Cross-Industry	212,069	280,059	372,989	567,659	27.9
Business: Vertical-Specific	634,921	683,817	736,543	863,662	8.0
Grand Total	*1,379,505*	*1,689,572*	*2,094,881*	*2,925,787*	*20.7*

The number is projected to rise beyond 2020, with an increased number of consumers purchasing more IoT devices, and businesses spending more to develop and maintain such devices. In 2017, in terms of hardware spending, the use of connected things among businesses was expected to drive 964 billion dollars in total across the cross-industry and vertical-specific business segments as shown in Table 12.2.

IoT Endpoint Spending for Consumer applications was expected to amount to 985 billion dollars in 2018. By the end of 2020, hardware spending from both segments is projected to reach nearly 3 trillion dollars. This predicted rise in the Internet of Things business will pave the way and define a mandate for optimizations of newer system and security technologies to make the future IoT devices more efficient and secure. We will next explore the secure storage and cloud technologies which may be leveraged to meet this mandate.

12.2.1 IoT Storage in the Cloud

The importance of data storage decisions stems from their implications in terms of application performance, data integrity, and data protection and restoration. The strategic decision to move from on-premise storage to cloud storage can be based on the initial costs, maintenance, type and amount of storage, that is, the Total Cost of Ownership (TCO) [4]. There are two main criteria that need to be kept in mind while selecting a storage solution, that is, security and cost. The nature of your data determines the best storage location for this data, such that this data can be accessed securely as required by the business without violating security and regulatory needs. The cost can be determined based on a detailed consumer needs (like expected latency and data availability) versus vendor study. This would include the analysis

of needs of different workloads and databases and their matched capabilities and costs with different cloud vendors.

In terms of system performance, hardware and software, system upgrades in the Cloud are much faster and inexpensive as compared to the months needed to upgrade on-premise storage. Disaster recovery in clouds is much more reliable in terms of the backup instances that are replicated in various physical locations, as compared to on-premise storage in tapes and disk backups. However, when it comes to security, technologists are conflicted between choices. The Snowden breach [5] in 2013 and the more recent Amazon Web Services (AWS) outage [6] in 2017 are two examples that prove that data is not secure whether on-premise or on the cloud.

The large volume and heterogenous types of data generated by the Internet of Things make the Cloud, with its high processing power, an ideal solution that some enterprises choose to process this data rather than build huge amounts of in-house capacity. Some examples of current cloud giants tackling IoT workloads are Microsoft's Azure IoT solution accelerators [7], Amazon's AWS IoT [8] and Google's Cloud IoT [9]. In general, cloud computing resources are fairly inexpensive in terms of availability and can also perform tasks rapidly. They easily adapt to the needs of each user that they serve, and the user location is generally irrelevant to the usage of the cloud from a technology perspective, but there sometimes could be regulatory implications. As long as you have the Internet, you can connect to the devices.

Currently, both IoT devices and the Cloud need designers and programmers to fix various incompatibilities to make sure they can work together better in the future. Figure 12.1 shows the layers involved with IoT data storage in the cloud. In the

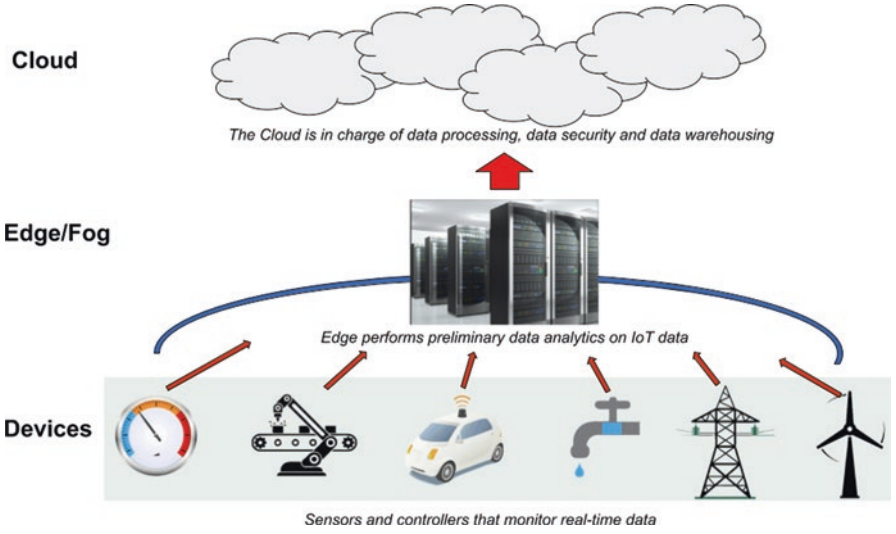

Fig. 12.1 The Cloud is an efficient storage solution for the massive amounts of data generated by the Internet of Things [10, 11]

future, some IoT devices will have increased device power to improve on-device data processing. Taking care of data processing and management on the device-level would mean that the Cloud resources can focus on things like data security, availability and storage. On the other hand, the Cloud deployments need to securely transfer and store data that are being communicated from different devices in different locations. In order to combine the flexibility and high computation power of the cloud with the intelligence of the IoT devices, new ways of analyzing and storing data is on the rise. This mitigates the dependency on cloud storage and enhances the role of IoT in performing computations and making decisions while staying closer to the end-user.

This new way is Edge [12] and Fog [13] computing. Instead of sending raw data off to the Cloud to be disseminated and analyzed, edge/fog computing is used for devices that require instantaneous or real-time decision making from their sensors in order for them to function correctly. An example of such a device is self-driving cars, which will create a new subsection of machine-to-machine communication in the form of vehicle-to-vehicle (V2V) communication. Any interactions with these cars will need to happen as close to real-time as possible. Edge computing moves data a far shorter distance, as compared to cloud, from the sensors themselves to local gateway device such as a switch or a router. This gateway device then performs the necessary processes and analysis and sends back decisions to the IoT device quicker than via cloud computing. Typically, this is done by the IoT devices transferring the data to a local device that includes compute, storage and network connectivity in a small form factor. Data is then processed at the edge, and all or a portion of it is sent to the central processing or storage repository in a corporate data center, co-location facility or Infrastructure as a Service (IaaS) cloud. Edge computing has various use cases. One is when IoT devices have poor connectivity and it is not efficient for IoT devices to be constantly connected to a central cloud. Other use cases involve latency-sensitive processing of information. Edge computing reduces latency because data does not have to traverse over a network to a data center or cloud for processing, making it ideal for situations where latencies of milliseconds can be untenable, such as in financial services or manufacturing. Fog computing is a term derived from edge computing. Fog refers to the network connections between edge devices and the cloud. Edge, on the other hand, refers more specifically to the computational processes being done close to the edge devices.

12.2.2 Security in the Internet of Things

The IoT industry at this time is in its infancy and has significant security and privacy implications. Matt Burgess mentions, “Everything that’s connected to the internet can be hacked, IoT products are no exception to this unwritten rule.” He further goes on to talk about the toy manufacturer Vtech, which lost the videos and pictures of children due to hackers compromising its insecure IoT systems [14]. Earlier, Wikileaks claimed that the CIA has been developing security exploits for a

connected Samsung television. The US director of national intelligence, James Clapper, predicted in 2016 that "In the future, intelligence services might use the Internet of Things for identification, surveillance, monitoring, location, tracking, and targeting for recruitment, or to gain access to networks or user credentials" [15].

In general, security is one of the biggest issues with the Internet of Things. Data collected by sensors may be extremely sensitive; for example, what we say or do at our own house, what we like and dislike, and our interests, to name a few. Security of such information is vital to consumer trust. But based on the track record provided by IoT devices, basic security concepts have been given little thought. The lack of patching capability in most IoT devices makes it hard to recover from software flaws detected on a regular basis. Hackers have actively begun targeting IoT devices such as routers and webcams because the inherent lack of security in these devices makes them easy to compromise and roll up into giant botnets. In 2017, researchers found 100,000 webcams originating in China that could be hacked with ease [16]. Some of these NeoCoolCam devices were priced as low as 39 dollars and were purchased all around the world. They contained improper quality assurance at the firmware level, several bugs affecting their authentication mechanisms and other buffer overflow vulnerabilities [17]. In 2017, other IoT vulnerabilities around the world have been brought to light like the internet-connected smartwatches for children, investigated by the Norwegian Consumer Council, that have been found to contain security vulnerabilities that allow hackers to track the wearer's location, eavesdrop on conversations, or even communicate with the user [18]. Currently, the tradeoff between cost and security has made the abovementioned problems widespread and intractable.

The Internet of Things bridges the gap between the digital world and the physical world, which means that hacking into devices can have dangerous real-world consequences. For example; hacking into the sensors controlling the temperature in a power station could trick the operators into making a catastrophic decision about incorrectly modulating the temperature even when not needed; taking control of a driverless car could lead to major accidents and risks of losing lives. In short, people should understand that there are many different use cases for the Internet of Things, many of which are yet to be explored, and that this has the potential to positively and negatively impact our lives. This chapter aims to throw light on some of the use cases and explore potential solutions to the problems we may face in the future due to the inherent properties of the Internet of Things.

12.3 Storage and the Internet of Things

IoT devices typically have limited data storage capabilities. Most of the data needs to be communicated using protocols such as Message Queuing Telemetry Transport (MQTT) [19] or Constrained Application Protocols (CoAP) [20], then further ingested by IoT services for additional processing and storage. MQTT is designed for connections with remote locations or where network bandwidth is limited,

whereas CoAP is used for constrained devices that communicate within the same constrained network. Dealing with the increased volume of data has made it difficult to secure the data in storage and to maintain integrity and privacy of the data. Apart from the obvious "more data means more storage" problem, there is also the added problem of dealing with different types of data generated by these devices. First, there is large-file data, such as images captured from medical devices. This data type is typically accessed sequentially. The second data type is very small, for example, log-file data captured by sensors. These sensors, while small in size, can create billions of files that must be accessed randomly. Determining the type of data to be stored is an essential first step to finding an optimized storage solution. The final goal is to build a multi-tiered storage solution that will work well with all types of data.

12.3.1 Existing Storage Technologies

Based on previously explained challenges with IoT data, storage solutions need to have three main properties: they need to securely store massive amounts of data, support horizontal scaling, and need to have the ability to deal with heterogeneous data resources. This section describes some existing storage technologies that are currently being applied to the Internet of Things.

Fazio et al. propose a two-layer hybrid architecture based on both SQL-like, and XML-like non-SQL technologies to provide a scalable, efficient and elastic sensing service [21]. They represent heterogeneous monitoring devices and data using Sensor Web Enablement (SWE) specifications, which defines data encodings and web services to store and access sensor-related data. Kang et al. propose a sensor-integrated radio frequency identification (RFID) data repository-implementation model using the MongoDB database [22]. They use a design based on horizontal data partitioning to maximize query speed and uniform data distribution over data servers. Gray et al. proposed a low complexity greedy distributed data replication mechanism to increase resilience and storage capacity of IoT based surveillance systems against node failure and local memory shortage [23]. Hu et al. introduce a ubiquitous data accessing method to deal with distributed storage of IoT based data in the healthcare area [24]. Teing et al. perform a forensic investigation of peer to peer (P2P) cloud storage for IoT networks using BitTorrent as a case study outlining strengths and weaknesses of using P2P cloud for IoT networks [25].

Jiang et al. talk about a database management model that combines multiple databases to store and manage structured IoT data [26]. They also propose a Representational State Transfer RESTful service generating mechanism to provide a hypertext transfer protocol (HTTP) interface for those applications that access the data stored based on their framework. Liu et al. propose a storage management solution based on NoSQL called IOTMDB [27]. Apart from handling large-scale data, this solution also tackles data sharing and collaboration. They also provide a query

mechanism to easily search and locate their shared data. Raj et al. go in a different direction as compared to literature around the time and exploit a document-oriented approach to propose a system that supports both heterogeneous and multimedia data [28]. They built their storage solution on top of the CouchDB database server and used a RESTful API to provide a rich set of features that targeted generic IoT applications. Distributed storage techniques came into the picture long before systems engineers started using secret- sharing to store their data.

Shyu et al. modify Shamir's secret-sharing [29] to utilize all coefficients in polynomials for larger data capacity at the data level [30]. Additionally, they use a distributed IoT storage infrastructure to provide scalability and reliability at the system level. Multiple IoT storage servers are aggregated to improve storage capacity, whereas individual servers can join and leave freely for flexibility at the system level. The importance of securely storing data is increasing as the amount of data produced by IoT devices is increasing. Satarkar et al. propose a secure storage architecture for IoT data, that combines Rivest–Shamir–Adleman (RSA) and AES to encrypt different files, of different sizes and contents [31]. Jararweh et al. simplify IoT management by proposing a software-defined based framework model to securely store data produced by IoT objects [32]. Shafagh et al. use blockchain, as an auditable and distributed access control layer on top of the storage layer, to enable secure and resilient access control management [33]. Their system accommodates for IoT data streams where streams are chunked, compressed, and encrypted in the application layer and authorized services are granted access only to the decryption keys.

12.4 Secret-Sharing

First invented by Adi Shamir and George Blakley in 1979, secret-sharing refers to all methods that can be used to distribute a secret among multiple participants in a group, such that each participant in the group only has a share of the secret and not the entire secret. The secret can be regenerated with the sufficient number of shares and individual shares are meaningless on their own. Figure 12.2 shows an overview of the secret-sharing technique. The regeneration of the secret is determined by a threshold scheme (T, N), where N is the total number of shares in the secret and T is the sufficient number of shares needed to regenerate the secret. Secret-sharing can be used as an alternative to traditional encryption techniques that tradeoff between confidentiality and reliability. The choice associated with storing an encryption key securely is either keeping a single copy of the key in one location for maximum confidentiality or having multiple copies of the key in different locations for maximum reliability. A single copy of the key for confidentiality allows for a single point of compromise and stealing that key would result in a storage system breach. Secret-sharing allows administrators to avoid key management issues. Having multiple copies of the key for reliability would allow an adversary to steal any of the copies which would lead to a breach of the storage system.

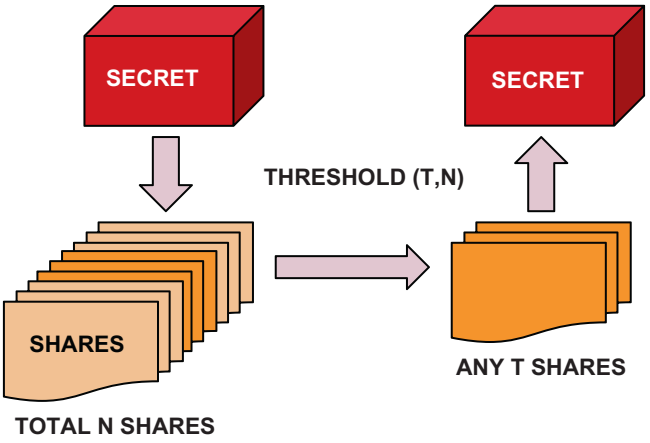

Fig. 12.2 Shamir's secret-splitting generates N equal-sized object shares out of which any T shares are both necessary and sufficient to rebuild the original object

12.4.1 Math Behind Secret-Sharing

The secret-sharing problem arises when a secret requires a certain number of participants to be mandatorily present before it can be revealed. Imagine a dying person who wants to leave his estate and wealth to his five children. He wants to make sure that the four older children do not end up bullying the youngest child to avoid giving him his share of riches. Therefore, the person makes a will stating their shares and locks it up. He splits the key into five parts such that the will can only be accessed when all five children are present and willing to unlock the box. We assume the will to be an integer S, that is the secret, the number of participants N or 5 in the case of this example and a minimum number of participants required for reconstruction T or 5 in the case of this example. The minimum number of participants required for reconstruction can be less than the value of N.

"The Mathematics of Secret Sharing" defines the math through a distributing algorithm and the reconstruction algorithm [34]. The distributing algorithm D accepts inputs S, N, T and produces an output list of N numbers $D(S, N) = \{x_1, x_2, \ldots, x_N\}$. These represent the secrets distributed to the N participants. The reconstruction algorithm R accepts T numbers $\{y_1, \ldots, y_T\}$ as input and gives an output number M. These algorithms are designed to hold two main properties:

- Knowledge of T or more shares makes S easily computable.
- Knowledge of less than T shares leaves S completely undetermined.

The main idea behind the secret sharing protocol is a polynomial interpolation. Polynomial interpolation states that given $k + 1$ points on a plane with x distinct values, there is a unique degree k polynomial. As a byproduct of this statement, there are infinitely many degree $k + 1$ polynomials that pass through the same points. The proof that such a polynomial definitely exists is divided into two parts, proof of

existence for the polynomial and proof of uniqueness for the polynomial. Refer to article "Proof of existence and uniqueness using splitting fields" for mathematical proofs of uniqueness and existence [35].

12.4.2 Types of Secret-Sharing

Secret-sharing has been built to provide a tradeoff between security and performance. The two main subclasses of secret-sharing are information-theoretically secure and computationally secure secret-sharing. Information-theoretically secure is when any number of shares less than the defined threshold is insufficient to generate the original data. The limitation of this type of secret-sharing is that each object share is of the same size, which means that the storage and transmission bandwidth required by the shares is equivalent to the size of the secret times the number of shares. In computationally secure secret sharing, shares are a fraction of a size of the secret. This uses repeated polynomial interpolation making the computations more complex and time-consuming, therefore allowing for data to be more secure. This can be used for secure information dispersal on the Web and in sensor networks.

Trivial secret-sharing has three types with threshold $T = 1$, $T = N$ and $1 < T \leq N$. For $T = 1$ trivial secret-sharing, the secret can be distributed to all N participants. For $T = N$ trivial secret-sharing, all shares of the secret are needed to reconstruct the secret. Trivial secret-sharing for $1 < T \leq N$ is where the complexity begins as we have to construct a secure secret-sharing scheme without needing all shares of the secret to rebuild the secret. Shamir came up with an information-theoretically secure secret-sharing scheme that uses Lagrange Interpolation and the size of each share does not exceed the size of the secret. The advantage of Shamir's secret-sharing is that keeping T constant, shares can easily be added or removed without affecting other shares. Proactive secret-sharing allows users to change threshold number with every update of the system but expects them to keep track of malicious users keeping expired shares. The verifiable secret-sharing scheme guarantees users that the other users in the group are not concealing or lying about the contents of their shares. Any of the above-mentioned types of secret-sharing can be used based on the type of data we choose to secure.

12.4.3 Applications of Secret-Sharing

Secret-sharing has been used as a method of secure information dispersal for a lot of different types of data. For example, for archival storage systems, it is preferable to use the information-theoretically secure secret-sharing over computationally-secure secret-sharing. This is because our adversary is assumed to have unlimited computation power and time, which means that given enough time he will eventually be able to compute the secret. However, a short-term data storage system can be secured by just using computationally secure secret-sharing.

Apart from archival storage secret-sharing has had many other applications (e.g., storing multimedia data). Shyu et al. talk about parallel implementations of Shamir's threshold secret-sharing scheme using sequential Central Processing Unit (CPU) and parallel Graphics Processing Unit (GPU) platforms to show that GPU can achieve a lucrative speedup over CPU when dealing with shared multimedia data [29]. Roy et al. have used (k, n) image secret sharing that involves sharing of a secret image into n number of pieces called shadow images in such a way that k or a greater number of shares can retrieve the original image [36]. They have proposed a (3, 4) image secret sharing scheme that has adopted the concept of visual cryptography over the 2X2 block. Security of the system is enhanced by scrambling the blocks using a pseudo-random sequence. Chen and Wu introduce a secure Boolean-based secret image sharing scheme which uses a random image generating function to generate a random image from secret images or shared images [37]. This efficiently increases the sharing capacity or storage bandwidth that is used to share the random image. Ching tackles Chen and Wu's inaccurate multi-secret image sharing (MSIS) by proposing a strong threshold (n, n) MSIS scheme without leaking partial secret information from $(n - 1)$ or fewer shared images. Komargodski et al. [38] construct a computational secret-sharing scheme for any monotone function in non-deterministic polynomial-time (NP) assuming witness encryption for NP and one-way functions. This results in a completeness theorem for secret-sharing where the computational secret-sharing scheme for any single monotone NP-complete function implies a computational secret-sharing scheme for every monotone function in NP. Huang et al. [39] propose secret sharing schemes to improve decoding bandwidth. Additionally, they consider the setting of secure distributed storage where the proposed communication efficient secret sharing schemes not only improve decoding bandwidth but further improve disk access complexity during decoding. Rawat et al. [40] talk about the centralized multi-node repair (CMR) model wherein multiple storage nodes can be reconstructed simultaneously at a centralized location, but there is a tradeoff between the amount of data stored and repair bandwidth. They provide another application of secret sharing in communication, where the codes for the multi-node repair problem are used to construct communication efficient secret sharing schemes with the property of bandwidth efficient share repair. Bai et al. propose a computationally secure and non-interactive verifiable secret sharing scheme that can be efficiently constructed from any monotone Boolean circuit [41]. Harn uses Lagrange's components, which are a linear combination of shares, to reconstruct a secret [42]. They extend their scheme to multi-secret sharing schemes as well. They compare existing multi-secret sharing schemes based on cryptographic assumptions like secure one-way function or solved discrete logarithm problem. Hadavi et al. present multiple partitioning methods that enable clients to efficiently search among shared secrets while preventing inference attacks on the part of data servers, even if they can observe shares and queries [43].

12.5 Secret-Sharing for the Internet of Things

Massive amounts of data from the Internet of Things are being stored on the cloud. Secret-sharing can be used with multiple deployments of the cloud to save cost while simultaneously improving the security of the data. Data can be grouped based on how frequently it is accessed or how secure it needs to be. The two kinds of deployments we talk about here are the hybrid cloud deployment and the multi-cloud deployment. Hybrid clouds can use a combination of private and public clouds to store all shares of data from IoT devices. Public clouds offer features such as scalability, low cost, and flexibility, while private clouds offer features like security, customization, enhanced control, and predictable costs. For example, if we were to look at an IoT network that consisted of devices that generated real-time data and data that needs to be stored long term, Figure 12.3 shows how the data may be distributed among multiple cloud instances.

For data that needs to be stored long-term, secret-sharing can be performed, and less than the threshold number of shares can be stored in a public cloud and any shares larger than that number can be stored in the private cloud. The main benefit of such a model would be that we can reduce the cost of storing an object by distributing shares between the private cloud and the cheaper public cloud. Additionally, our objects will be secured as part of the shares that are in the more secure private cloud. In case of a breach in the public cloud, shares stolen by an adversary would be less than the threshold number and therefore not useful in rebuilding the object (we are assuming here that private cloud is more secure than public cloud). A multi-cloud deployment can be used to prevent users from trusting a single cloud provider to store their data securely. Such a model could enhance data privacy by distributing object shares among multiple cloud providers, and may also be fault tolerant towards server downtimes or failures. Every provider would have shares of the data, but not a sufficient number of shares needed to rebuild the original data object. This model can be used to store any generated data that does not need to be immediately sent to the devices as feedback.

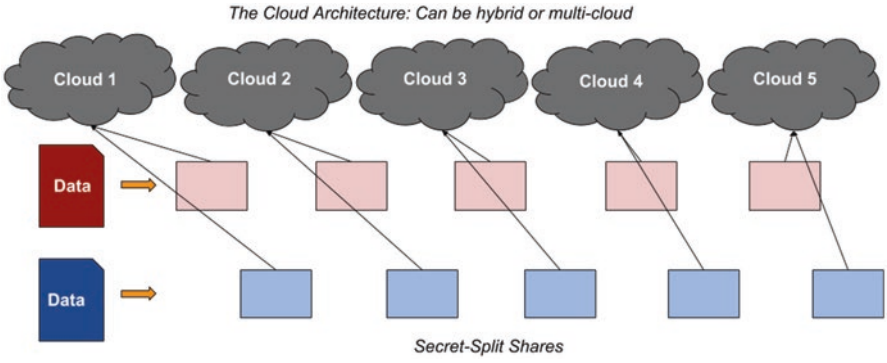

Fig. 12.3 IoT data storage using hybrid cloud or multi-cloud architectures

12.6 Conclusion

Storage systems for data produced by the Internet of Things need to provide properties like data security, availability, and scalability, and also be able to handle heterogeneous types of data. Cloud storage is an efficient way to store the massive amounts of data being generated by IoT devices as it has provisions for all the above properties. This chapter highlights issues with current storage techniques for IoT data and suggests secret-sharing as an efficient way to securely store such data on the Cloud. Secret-sharing is a distributed storage technique that not only stores data securely but also provides data availability. Additionally, the different types of secret-sharing techniques (information-theoretically secure and computationally secure) make it an ideal solution for both long- and short-term storage of data. In this chapter, we hypothesize about two secure distributed cloud storage models- a hybrid cloud model and a multi-cloud model, each of which has different advantages (cost versus availability). Both models may perform computations on gateway devices and therefore can work with IoT devices that have a wide range of computing power. Our proposed storage models could prove to be extremely efficient and secure for the recently rising IoT ecosystems used to run smart cities and smart hospitals.

References

1. Hybrid Cloud Architecture. https://aws.amazon.com/enterprise/hybrid/
2. Hung M. Leading the IoT, Gartner insights on how to lead in a connected world. https://www.gartner.com/imagesrv/books/iot/iotEbook_digital.pdf
3. Meulen R. Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016: https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016
4. Scheier B. Decision guide: public cloud versus on-premise storage. https://www.hpe.com/us/en/insights/articles/decision-guide-public-cloud-versus-on-prem-storage-1701.html
5. Snowden E. Leaks that exposed US spy program. https://www.bbc.com/news/world-us-canada-23123964
6. Massive Amazon Cloud service outage disrupts sites. https://www.usatoday.com/story/tech/news/2017/02/28/amazons-cloud-service-goes-down-sites-scramble/98530914/
7. Azure IoT solution accelerators. https://azure.microsoft.com/en-us/features/iot-accelerators/
8. AWS IoT. https://aws.amazon.com/iot/
9. Google Cloud IoT. https://cloud.google.com/solutions/iot/
10. Mainframes and supercomputers, from the beginning till today. http://www.cpushack.com/2018/05/27/mainframes-and-supercomputers-from-the-beginning-till-today/
11. 4EdgeComputingTechnologiesEnablingIoT-ReadyNetworkInfrastructure.https://www.lanner-america.com/blog/4-edge-computing-technologies-enabling-iot-ready-network-infrastructure/
12. Edge Computing. https://en.wikipedia.org/wiki/Edge_computing
13. Fog Computing. https://en.wikipedia.org/wiki/Fog_computing
14. Burgess M. What is the Internet of Things, WIRED explains. https://www.wired.co.uk/article/the Internet of Things-what-is-explained-iot
15. Timm T. US Intelligence Chief: We might use the Internet of Things to spy on you. https://www.theguardian.com/technology/2016/feb/09/the Internet of Things-smart-home-devices-government-surveillance-james-clapper

16. Ranger S. What is IoT? Everything you need to know about the Internet of Things right now. https://www.zdnet.com/article/what-is-the-the Internet of Things-everything-you-need-to-know-about-the-iot-right-now/
17. 175,000 IoT cameras can be remotely hacked thanks to flaws, says security researcher. https://www.zdnet.com/article/175000-iot-cameras-can-be-remotely-hacked-thanks-to-flaw-says-security-researcher/
18. Security flaws in children's smartwatches make them vulnerable to hackers. https://www.zdnet.com/article/security-flaws-in-childrens-smartwatches-make-them-vulnerable-to-hackers/
19. MQTT Wikipedia. https://en.wikipedia.org/wiki/MQTT
20. Constrained Application Protocol Wikipedia. https://en.wikipedia.org/wiki/Constrained_Application_Protocol
21. Fazio M, Celesti A, Villari M, Puliafito A (2014) The need of a hybrid storage approach for IoT in PaaS cloud federation, 28th international conference on advanced information networking and applications
22. Kang YS, Park IH, Rhee J, Lee YH (2016) MongoDB-based repository design for IoT-generated RFID sensor big data. IEEE Sensors J. https://doi.org/10.1109/JSEN.2015.2483499
23. Gray V, Gonizzi P, Ferrari G, Leguay J (2015) Data dissemination scheme for distributed storage for IoT observation systems at large scale. Inf Fusion. https://doi.org/10.1016/j.inffus.2013.04.003
24. Hu J, Xu B, Xu LD, Bu F (2014) Ubiquitous data accessing method in Iot-based information system for emergency medical services. IEEE Trans Ind Inform. https://doi.org/10.1109/TII.2014.2306382
25. Teing YY, Dehghantahna A, Yang L (2017) Forensic investigation of P2P cloud storage services and backbone for IoT networks: BitTorrent sync as a case study. J Comput Electr Eng. https://doi.org/10.1016/j.compeleceng.2016.08.020
26. Jiang Z, Jiang L, Xu LD, Xu B (2014) An IoT-oriented data storage framework in cloud computing platform. IEEE Trans Ind Inform. https://doi.org/10.1109/TII.2014.2306384
27. Liu Y, Li T, Mao W (2012) A storage solution for massive IoT data based on NoSQL. IEEE international conference on green computing and communications
28. Raj M, Francesco MD, Li N, Das SK (2012) A storage infrastructure for heterogeneous and multimedia data in the internet of things. IEEE international conference on green computing and communications
29. Shyu SJ, Tsai YZ (2018) Shamir's secret sharing scheme in parallel. In: Peng SL, Wang SJ, Balas V, Zhao M (eds) Security with intelligent computing and big-data services. SICBS 2017. *Advances in Intelligent Systems and Computing* (vol. 733). Springer, Cham
30. Chen V, Jiang H, Shen F, Jeong YJ (2015) A secure and scalable storage system for aggregate data in IoT. Future Gener Comput Syst. https://doi.org/10.1016/j.future.2014.11.009
31. Satarkar P, Bokefode J, Bhise A, Modani D (2016) Developing a secure cloud storage system for storing iot data by applying role-based encryption. Procedia Comput Sci. https://doi.org/10.1016/j.procs.2016.06.007
32. Jararweh Y, Al-Ayyoub M, Darabseh A, Benkhelifa E, Vouk M, Rindos A (2015) SDIoT: a software defined based Internet of Things framework. J Ambient Intell Humaniz Comput 6:453–461
33. Shafagh H, Burkhalter L, Hithnawi A, Duquennoy S (2017) Towards blockchain-based auditable storage and sharing of IoT data. Proceedings of the 2017 on cloud computing security work-shop
34. Chadha K. The mathematics of secret-sharing. https://jeremykun.com/2014/06/23/the-mathematics-of-secret-sharing/
35. NTNU. Proof of existence and uniqueness using splitting fields. https://wiki.math.ntnu.no/_media/ma3202/2015v/ch16-splittingfields.pdf
36. Roy R, Bandhopadhyay S, Kandar S, Dhara BC (2015) A novel 3–4 image secret sharing scheme. International conference on advances in computing, communications and informatics

37. Chen CC, Wu WJ (2014) A secure Boolean-based multi-secret image sharing scheme. J Syst Softw. https://doi.org/10.1016/j.jss.2014.01.001
38. Komargodski I, Naor M, Yogev E (2016) Secret-sharing for NP. J Cryptol 30:444–469
39. Huang W, Langberg M, Kliewer J, Bruck J (2016) Communication efficient secret sharing. IEEE Trans Inf Theory 62:7195–7206
40. Rawat AS, Koyluoglu OO, Vishwanathan S (2016) Centralized repair of multiple node failures with applications to communication efficient secret sharing. Cornell University Library
41. Bai G, Damgard I, Orlandi C, Xia Y (2016) Non-interactive verifiable secret sharing for monotone circuits. Springer International Conference on Cryptology
42. Harn L (2013) Secure secret reconstruction and multi-secret sharing schemes with unconditional security. Secur Commun Netw. https://doi.org/10.1002/sec.758
43. Hadavi MA, Jalili R, Damiani E, Cimato S (2015) Security and searchability in secret sharing-based data outsourcing. Int J Inf Secur 14:513–529