IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF VIRGINIA
Alexandria Division

SECURITY FIRST INNOVATIONS,
LLC.,

Plaintiff,

v.

INTERNATIONAL BUSINESS
MACHINES CORPORATION,

Defendant.

Civil Action No. ________________

JURY TRIAL DEMANDED

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Security First Innovations, LLC ("SFI") files this complaint for patent infringement pursuant to 35 U.S.C. §§ 100 *et seq.* against Defendant International Business Machines Corporation ("IBM") for infringement of U.S. Patent Nos. 9,135,456 (the "'456 Patent"), 8,904,194 (the "'194 Patent"), and 8,271,802 (the "'802 Patent") (collectively, the "Asserted Patents"; attached as Exhibits A, B, and C, respectively) and alleges as follows:

NATURE OF THIS ACTION

1. This is an action for patent infringement arising under 28 U.S.C. §§ 1331 and 1338(a) and the United States Patent Act, 35 U.S.C. §§ 100 *et seq*, including 35 U.S.C. §§ 271, 281, 283, 284, and 285. SFI seeks damages and other appropriate relief for Defendant's widespread and willful infringement of the Asserted Patents such as by IBM's Cloud Object Storage System. As a direct result of its infringement, IBM has received billions of dollars from its cloud computing customers who value the enhanced data security provided by IBM's unauthorized use of SFI's technology claimed in the Asserted Patents.

THE PARTIES

2. Plaintiff SFI is a limited liability company duly organized and existing under the laws of the Commonwealth of Virginia. SFI is located at 44095 Pipeline Plaza, Suite 140, Ashburn, Virginia 20147.

3. On information and belief, Defendant IBM is a corporation duly organized and existing under the laws of the State of New York, with its corporate headquarters at 1 Orchard Road, Armonk, NY 10504.

JURISDICTION AND VENUE

4. This Court has subject matter jurisdiction over this patent infringement action pursuant to 28 U.S.C. §§ 1331 and 1338(a) because this action arises under the patent laws of the United States, Title 35 United States Code, including 35 U.S.C. § 100 *et seq*. This complaint includes claims for patent infringement arising under the patent laws of the United States, including 35 U.S.C. § 271 *et seq*.

5. This Court has personal jurisdiction over IBM because IBM makes, uses, offers for sale, and/or provides products and services in the Eastern District of Virginia, has committed and continues to commit acts of infringement in this District, owns or rents real estate in the District, including by maintaining a physical presence at multiple office locations and data centers in this District, and/or has engaged in continuous and systematic activities in this District. On information and belief, IBM derives substantial revenue from the acts of infringement in the Eastern District of Virginia and derives substantial revenue from interstate and international commerce associated with the infringing products.

6. Venue is proper in this judicial District pursuant to 28 U.S.C. §§ 1391(b) and (c) and 35 U.S.C. § 1400(b) at least because IBM has committed and continues to commit acts of infringement within this District giving rise to this action and has a regular and established place

of business in this District, including office spaces and data centers located at least in the cities of Arlington, Ashburn, Chantilly, Fairfax, Herndon, Reston, and Sterling. IBM also provided and continues to provide infringing products and/or services to residents, businesses, and government agencies located in this District.

7. The relevant technology in this case includes the encryption and data parsing technology that, on information and belief, has been implemented in IBM Cloud, including IBM Cloud Object Storage, on servers in IBM's data centers located in this District. Due in part to IBM's extensive presence in Northern Virginia, and the uniquely significant role that its data centers play in the infringing systems, the Eastern District of Virginia has a strong interest in this case.

8. IBM also maintains a significant employee presence in the District. On information and belief, IBM employs more than 650 employees in this District, not including employees that IBM acquired through acquisitions like Octo, as described below. As of March 18, 2025, IBM has listed job postings for at least 52 on-site or hybrid positions based in this District on LinkedIn, including postings for Cloud Engineers, Data Scientists, Data Analysts, and Software Engineers.

9. Beyond its relevant employee presence, based on information and belief, IBM maintains at least five data centers in the District: WDC01 (Chantilly); WDC03 (Ashburn); WDC04 (Ashburn); WDC06 (Ashburn); and WDC07 (Ashburn).

10. IBM's data centers play a critical and uniquely relevant role to the infringing systems and products that IBM offers to its customers, including those customers located in this District. IBM has significantly invested in cloud technology. For example, in 2014, IBM reportedly spent $2 billion on acquiring SoftLayer, a cloud infrastructure service, and another $1.2 billion investment to expand SoftLayer's global data footprint to, among other things, "add[]

capacity in Dallas, San Jose, Seattle and Virginia."[1] And this infrastructure is critical to IBM's infringing systems and products. As IBM explains, "deployments of IBM Cloud Object Storage System solution [(*i.e.*, the infringing system)] can span multiple data centers."[2]

11. IBM has also acquired relevant companies in this District. For example, in 2022, IBM acquired Octo, an "IT modernization and digital transformation services provider" that provides services for the U.S. federal government, including defense, health, and civilian agencies.[3] Among other services that Octo provides to its customers, it also provides, among other things, "Cloud and Infrastructure, Data Management and Analytics, [and] Cybersecurity."[4] As of at least 2022, Octo employed 1,500 employees.[5]

12. In addition to the Eastern District of Virginia being a proper venue at the heart of this controversy, SFI is also at home in the District. SFI is a Virginia Domestic Limited Liability Company headquartered at 44095 Pipeline Plaza, Suite 140, Ashburn, VA 20147.

[1] *IBM SoftLayer: One Year After the Acquisition*, DATA CENTER KNOWLEDGE (July 14, 2014) https://www.datacenterknowledge.com/deals/ibm-softlayer-one-year-after-the-acquisition.

[2] IBM Cloud Object Storage System Version 3.18.3: Definitive Guide to Dispersed Storage at 20, IBM (2024), https://www.ibm.com/docs/en/STXNRM_3.18.3/pdf/coss_dispersed_book.pdf [IBM Definitive Guide]; *see also* Vasfi Gucer et al., *IBM Cloud Object Storage System Product Guide* at 3, IBM (June 2023), https://www.redbooks.ibm.com/redbooks/pdfs/sg248439.pdf [IBM Product Guide].

[3] *IBM to Acquire Octo*, IBM (Dec. 7, 2022), https://newsroom.ibm.com/2022-12-07-IBM-to-Acquire-Octo.

[4] *Octo is Awarded IT Infrastructure and Operations Call Order to Support National Cancer Institute*, BUSINESSWIRE (July 25, 2023), https://www.businesswire.com/news/home/20230725438672/en/Octo-is-Awarded-IT-Infrastructure-and-Operations-Call-Order-to-Support-National-Cancer-Institute.

[5] *IBM to Acquire Octo*, IBM (Dec. 7, 2022), https://newsroom.ibm.com/2022-12-07-IBM-to-Acquire-Octo.

FACTUAL BACKGROUND

A. Data Security & Encryption

13. The security of the important, valuable, private, or even simply personal has been a primary concern for thousands of years—from treasure keeps securing gold, to bank vaults securing currency, to home intruder systems securing personal belongings. As the nature of what is being secured has changed, so too have the methods of security.

14. Today, as more and more information is stored online through the use of computer technology, information is now at the top of the list of things that needs securing. Generally known as "data security," the protection of information is of the utmost importance to those seeking to safeguard their important, valuable, private, and /or personal information. And just as physical security has evolved, so too has data security.

15. At a high level, data can be sorted into two categories: Data At-Rest, and Data In-Transit. Data At-Rest refers to data that is not actively moving through networks (*e.g.*, stored), and Data In-Transit refers to data that is actively moving from one location to another, such as between users or devices through the internet or through a private network. A popular way of securing each category of digital data is called encryption. To that end, Encryption At-Rest refers to the encrypting of Data At-Rest, and Encryption In-Transit refers to encrypting Data In-Transit.

16. Cloud storage, such as is used in IBM's Cloud Object Storage System, is a method of computer data storage where, instead of keeping data on local devices, data storage is outsourced to third parties called cloud storage providers. Generally, the data is stored using physical servers that are operated by a third party like IBM. And cloud customers can generally access that data from the cloud storage provider through the internet. Encryption At-Rest is a critical component of any cloud storage system, and employing Encryption At-Rest in such systems comes with many

challenges, including a number that are overcome by the systems and methods claimed in the Asserted Patents.

17. In general, encryption works by transforming data from human-readable "plain text" to unreadable "ciphertext" using a specific algorithm and a cryptographic key. The ciphertext is indecipherable without the cryptographic key, which is needed to transform the ciphertext back into a plaintext format. As IBM explains it: "Encryption works by using encryption algorithms to scramble data into an indecipherable format. Only the authorized parties with the right secret key . . . can unscramble the data."[6]

18. Encryption serves several important functions. For example, encryption is not only used to protect data (*e.g.*, from malevolent actors who cannot access it without the cryptographic key), but it also can be used for verification/authentication purposes (*e.g.*, if the right key cannot decrypt the data, the data may have been altered).

19. As a result, safely storing and keeping track of encryption keys—so-called "key management"—is critical for effective data encryption. As IBM explains:

> To understand why, consider the example of a safe. If an individual forgets their code to a safe or it ends up in the wrong hands, they risk losing access to their most valuable possessions or having them stolen. The same logic applies to cryptographic keys. If organizations don't properly manage their keys, they can lose the ability to decrypt and access data or expose themselves to data breaches. For this reason, organizations often prioritize investing in key management systems. These services are critical given that organizations frequently manage a complex network of cryptographic keys, and many threat actors know where to look for them.[7]

[6] *What Is Encryption?*, IBM, https://www.ibm.com/think/topics/encryption (last accessed March 23, 2025).

[7] *Id.*

20. While encryption is a critical step in securing Data At-Rest, it may not be sufficient, especially as technology advances and malicious actors become more and more sophisticated. As IBM itself recognizes, encryption alone may be vulnerable to, among others, quantum computing; brute-force attacks; algorithm vulnerabilities; side-channel attacks; and inadequate key management.[8] Cloud computing providers like IBM try to address these vulnerabilities by pairing encryption of Data At-Rest with other techniques, such as obfuscation, that render the data less accessible to an attacker or that make it impossible to reconstruct even if encryption fails.

21. The Asserted Patents describe and claim a novel technology that provides reliable key management and obfuscation techniques that greatly increase the security of Data At-Rest as compared to the methods and systems that were previously available. The patented technology was developed by Security First Corporation ("SFC") beginning in the early 2000s, and was disclosed to IBM repeatedly over the next 15 years. IBM took that technology without SFC's permission and without compensating SFC, and infringes the Asserted Patents by using the same technology today to secure its own and its customers' Data At-Rest, such as in its Cloud Object Storage System (the exemplary "Accused Product").

B. Security First Corporation

22. SFC was established in 2002 to develop innovative data security systems.

23. From its inception, SFC focused on improving data security. It started with biometric data, but then shifted to data stored in cloud computing environments, realizing that the state of security in cloud computing was woefully inadequate. Beginning in around 2004, SFC developed technology to substantially improve the security of both encrypted and unencrypted Data At-Rest by, among other things, unconventionally parsing and splitting the data into different

[8] *Id.*

portions before storage, and storing those portions in different locations. The Asserted Patents disclose and claim this technology, and its inventors were members of the original founding team behind SFC. The technology claimed in the Asserted Patents is the result of many years of effort to develop a multi-layered data security system that refined and combined the concept of splitting data with the concept of data encryption.

24. Over the years, SFC created a number of products offering data security solutions based on this technology, including, for example, the SPxSHARC, SPxGateway, and SPxClient. These products were a part of SFC's suite of end-to-end solutions for data security—protecting point of sale, enterprise, cloud, networks, and more. SFC's technology has also been licensed and used by several companies to improve their data security, including, for example, Unisys.

25. Plaintiff SFI was founded by the longtime former chairman of SFC. In 2022, SFI acquired the patents that were developed by SFC, including the Asserted Patents. Today, SFI is their sole assignee.

C. The Claimed Technology

26. The inventions claimed in the Asserted Patents relate to systems for securing Data At-Rest from unauthorized access or use. *See, e.g.*, '194 Patent, 1:17-18.[9] More specifically, the inventions claimed in the Asserted Patents provide, among other things, solutions to common problems relating to prior art data security systems.

27. The Asserted Patents explain that because "individuals and businesses conduct an ever-increasing amount of activities on and over computer systems" and "[t]hese computer

[9] The '194 Patent and the '802 Patent contain substantially identical specifications. The '456 Patent's specification is substantially similar but includes two additional paragraphs. *Compare* '456 Patent, 75:1-53 (adding 75:7-39), *with* '194 Patent, 75:1-21, *and* '802 Patent, 74:62-75:14. For administrative ease, citations to the common disclosures in the specification will be to the '194 Patent.

systems . . . are often storing, archiving, and transmitting all types of sensitive information," there is "an ever-increasing need . . . for ensuring data stored and transmitted over these systems cannot be read or otherwise compromised." '194 Patent, 1:22-29.

28. As previewed above, encryption (or cryptography, more generally) is one common method for providing enhanced security. And there are many different ways for encrypting data. For example, as the Asserted Patents explain, "[o]ne popular cryptography system is a public key system that uses two keys, a public key known to everyone and a private key known only to the individual or business owner thereof. Generally, the data encrypted with one key is decrypted with the other and neither key is recreatable from the other." '194 Patent, 1:44-49. Another example is to include biometrics as part of the cryptographic system's authentication process—like using a thumb print or other measurable physical characteristic that can be checked by an automated system. '194 Patent, 2:4-14. And often, in such systems, a user's biometric data is "stored on mobile computing devices, such as, for example, a smartcard, laptop, . . . or mobile phone." *Id.*

29. But these prior art cryptographic systems have several flaws. As one example, both systems are highly user-reliant. '194 Patent, 1:50-52; *id.*, 2:15-26.

30. With respect to traditional cryptographic systems, as the Asserted Patents explain, "[u]nsophisticated users [] generally store the private key on a hard drive accessible to others through an open computer system," like the internet, or "may choose poor names for files containing their private key," which makes the key or keys "susceptible to compromise." *Id.*, 1:53-60. Additionally, a user "may save his or her private key on a computer system configured with an archiving or backup system, potentially resulting in copies of the private key traveling through multiple computer storage devices or other systems," causing a security vulnerability referred to as "key migration." *Id.*, 1:61-66. Relatedly, "many applications provide access to a user's private

key through, at most, simple login and password access," which "provide little security" and involve a higher administrative burden. *Id.*, 1:66-2:3, 1:30-36.

31. The Asserted Patents explain that biometric cryptographic systems suffer from similar drawbacks.

> For example, the mobile user may lose or break the smartcard or portable computing device, thereby having his or her access to potentially important data cut-off. Alternatively, a malicious person may steal the mobile user's smartcard or portable computing device and use it to effectively steal the mobile user's digital credentials. On the other hand, the portable-computing device may be connected to an open system, such as the Internet, and, like passwords, the file where the biometric is stored may be susceptible to compromise through user inattentiveness to security or malicious intruders.

Id., 2:16-26.

32. The Asserted Patents describe and claim solutions to these problems. For example, the '456 Patent is directed to a particular storage method that involves encrypting a first set of data using an encryption key, logically combining a portion of the encrypted data with the encryption key to create a resultant that is part of the second data set, creating redundancy information based on that second set of data, using a secure data parser to split the second data set into multiple portions (*i.e.*, the plurality of shares), and storing a plurality of those shares and the redundancy information wherein some of those shares are stored in separate storage locations. '456 Patent, claim 1.

33. Before the invention disclosed and claimed in the '456 Patent, it was wholly unconventional to "logically combine" the encrypted data with the encryption key to produce a resultant or to include data "indicative of [] the encryption key." This would have been regarded as creating a vulnerability or insecurity by co-locating encrypted data with information that might facilitate decrypting such data. Despite that, however, some of the claimed inventions unconventionally store data indicative of at least one encryption key with the data shares, which

provides certain unexpected benefits. Among other advantages, this reduces user-reliance such that the keys "are not lost, stolen, or compromised, thereby advantageously avoiding a need to continually reissue and manage new keys and authentication data." *See* '456 Patent, 3:10-14. As another example, claim 1 of the '456 Patent unconventionally requires "storing the plurality of [] shares and the redundancy information" in "separate storage locations and the first data set is recoverable using at least a threshold number, less than n, of the plurality of n shares." These features, in particular, enable the invention to be deployed effectively in a cloud environment.

34. The '802 Patent is directed to a method for splitting the encrypted data set involving, among other things, generating data splitting information that determines into which of a plurality of shares a unit of data will be placed, separating the encrypted data into those shares, and including in the plurality of shares data indicative of the encryption key and integrity information. '802 Patent, claim 1.

35. Before the inventions disclosed and claimed in the '802 Patent, it was unconventional to distribute split data and the data indicative of the encryption key across multiple different storage devices or facilities. This would have required, at a minimum, an additional mechanism to locate the split data shares across the different storage devices and then reassembling the data shares from the different storage devices in order to reconstitute the original data set. As yet another example, it was also unconventional to reconstruct the first data set using a threshold number of the split data shares that is less than the total number of shares—as recited in, *e.g.*, '456 Patent, claim 1 and '802 Patent, claim 1.

36. The '194 Patent is directed to a method for retrieving and reconstructing such data, involving, among other things, identifying from the plurality of storage locations the fastest-

responding storage devices necessary to retrieve the minimum number of shares to reconstruct the data set. '194 Patent, claim 1.

37. Prior to the inventions claimed in the '194 Patent, it was unconventional to identify the minimum threshold number of those split data shares and retrieve them from the fastest-responding storage devices necessary to obtain that number. *E.g.*, '194 Patent, claim 1. This would have required additional mechanisms to identify and reassemble the data shares from the storage devices in order to reconstitute the original data set, as well as to identify the fastest-responding storage devices. Claim 1 of the '194 Patent also unconventionally requires "storing the plurality of shares at a plurality of storage devices."

38. In addition to providing solutions to the problems described above, the Asserted Patents' claims are directed to improving the basic functionality of a computer system—by improving its ability to securely store information and by improving the efficiency of such storage. The '456 Patent's claims, for example, require that the encryption key be "logically combin[ed]" with the first data set to produce a resultant, which, together with the first data set, constitutes the second data set that is split and distributed among storage locations. *E.g.*, '456 Patent, claim 1. Similarly, the '802 Patent's claims require the inclusion of data indicative of the encryption key with the plurality of shares. *E.g.*, '802 Patent, claim 1. Storing such data indicative of the encryption key (like the resultant) improves the efficiency of the decryption process because the information needed to identify or locate the encryption key is locally available with the data itself.

39. As a further example, the Asserted Patents' claims are directed to other improvements in securing data in the specific context of cloud storage where many different storage locations may be used. For example, through the disclosed and proprietary system and because of the redundancy, the first set of data can be reconstructed using a minimum number of

shares (less than the total) that have been retrieved from the fastest-responding storage devices. *E.g.*, '194 Patent, claim 1; '802 Patent, claim 1. Advantageously, this means that for data shares that are geographically separated to ensure their security, the data set may be reconstructed faster—even when certain data storage locations are offline for maintenance or other reasons, so long as the minimum threshold of shares is obtained. *E.g.*, '194 Patent, 18:47-19:5.

40. Accordingly, the claimed inventions of the Asserted Patents improve the functions of a computer system and improve upon conventional cryptographic security methods for at least the reasons set forth above.

41. In addition, the Asserted Patents' claims are necessarily rooted in computer technologies and provide technical and practical solutions to overcome problems associated with prior art cryptographic systems. The claimed systems and methods are rooted in computer technologies at least because they are directed to improving security of data storage in a computer system. The Asserted Patents address problems that arise only in the context of computer systems and the use of networked storage systems.

42. For at least the foregoing reasons, the elements of the Asserted patents, individually or as part of an ordered combination, cover non-routine, unconventional, inventive features that provide specific technical and practical improvements to solve a particular problem in the field of information security.

D. The Exemplary Accused Product: IBM Cloud Object Storage

43. IBM offers a suite of cloud services generally referred to as IBM Cloud. On information and belief, IBM began offering cloud-related services in 2007.[10] Then, on information

[10] *The History of IBM Cloud: A Journey Through Transformation*, REDRESS COMPLIANCE (January 16, 2025) https://redresscompliance.com/ibm-cloud-history-a-journey-through-transformation/#Introduction_To_The_History_of_IBM_Cloud.

and belief, to offer additional cloud services, IBM launched SmartCloud in 2011 and subsequently acquired SoftLayer Technologies in 2013.[11] IBM later rebranded its cloud-based services as IBM Cloud.[12]

44. Among other services, IBM Cloud includes IBM Cloud Object Storage, which "stores encrypted and dispersed data across multiple geographic locations."[13] IBM Cloud Object Storage touts the use of an "innovative approach for cost-effectively storing large volumes of unstructured data that ensures security, availability, and reliability."[14] IBM states that this security is achieved by using an "Information Dispersal Algorithms (IDA) to separate data into unrecognizable 'slices'" and that the "slices" are distributed "across a network of data centers, making transmission and storage of data inherently private and secure."[15] Additionally, as IBM explains, "[n]o complete copy of the data resides in any single storage node, and only a subset of nodes needs to be available to fully retrieve the data on the network."[16]

45. In IBM's Cloud Object Storage System "[a]ll data is encrypted at rest by default."[17] It includes three primary components: the "IBM Cloud Object Storage Manager"; the "IBM Cloud

[11] *Id.*

[12] *Id.*

[13] *Getting Started with IBM Cloud Object Storage*, IBM (last updated September 25, 2024) https://cloud.ibm.com/docs/cloud-object-storage?topic=cloud-object-storage-getting-started-cloud-object-storage.

[14] *Data Security*, IBM (last updated April 17, 2024) https://cloud.ibm.com/docs/cloud-object-storage?topic=cloud-object-storage-security.

[15] *Id.*

[16] *Id.*

[17] Chris de Almeida, et al, *IBM Cloud Object Storage Concepts and Architecture: System Edition* at 30 (2023), https://www.redbooks.ibm.com/redpieces/pdfs/redp5537.pdf [IBM Red Paper]; *see*

Object Storage Accesser"; and the "IBM Slicestor."[18] The IBM Cloud Object Manager provides a "management interface that is used for administrative tasks, such as system configuration."[19] The IBM Cloud Object Accesser "encrypts and encodes data on write," "decodes and decrypts it on read," and "transforms data by using an IDA."[20] And the IBM Cloud Object Storage Slicestor is "responsible for storing the data slices. It receives data from the Accesser node."[21]

46. Employing the components discussed above, at a high level, IBM's Cloud Object Storage "uses three steps for slicing, dispersing, and retrieving data."[22] In the first step, "[d]ata is virtualized, transformed, sliced and dispersed using IDAs." In the second step, "[s]lices are distributed to separate disks, storage nodes and geographic locations." And in the third step, "[t]he data is retrieved from a subset of slices."[23] IBM provides the following exemplary diagram depicting these steps.[24]

also Data Security, IBM (last updated April 17, 2024) https://cloud.ibm.com/docs/cloud-object-storage?topic=cloud-object-storage-security.

[18] IBM Red Paper at 6.

[19] *Id.*

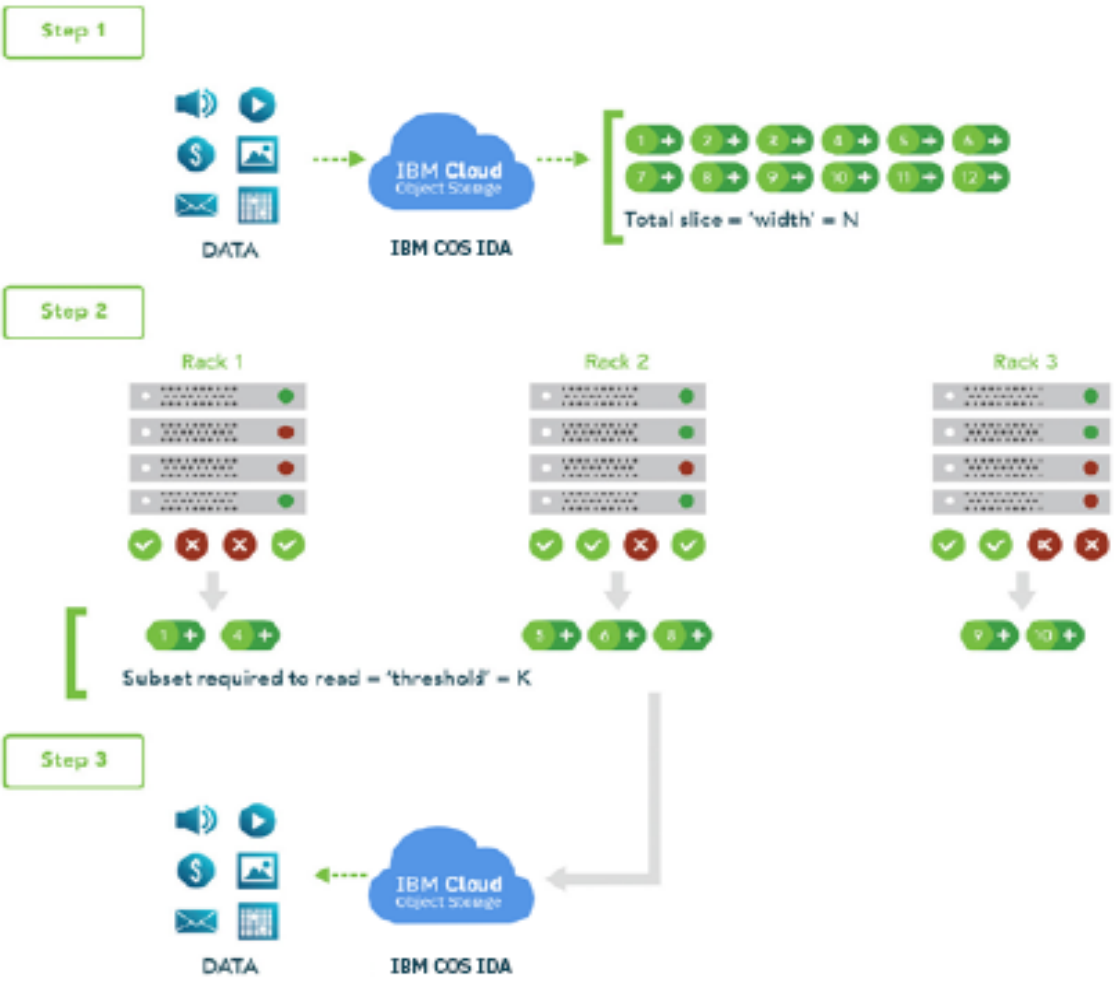
[20] *Id.*

[21] *Id.*

[22] IBM Definitive Guide at 5.

[23] *Id.*

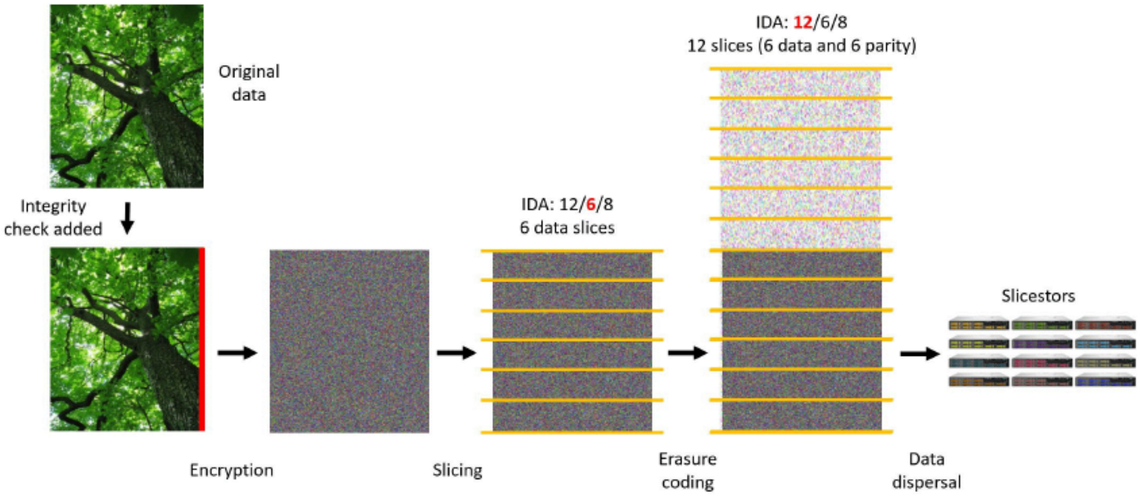
[24] *IBM Cloud White Paper: The Definitive Guide to Cloud Object Storage System Dispersed Storage* at 6, IBM, (Oct. 2017) [IBM White Paper].



47. Using the technology claimed in the Asserted Patents, IBM's Cloud Object Storage System encrypts the data that it seeks to protect, then calculates the hash of the encrypted data, which it uses to "[c]alculate [the] exclusive-OR (XOR) of [the] hash and encryption key" and "[a]ppend[s] the result to the encrypted data."[25] The data is then sliced and redundancy

[25] IBM Red Paper 17-18.

information is created, as depicted in the exemplary diagram below.[26] Then, when necessary, as IBM explains it, "[t]he data is retrieved from a subset of slices."[27]



48. As explained below, IBM's Cloud Object Storage System infringes the Asserted Patents, both directly and indirectly.

49. IBM offers its Cloud Object Storage System service to subscribers who use the service to store data on systems configured with Cloud Object Storage System software, where the systems may be owned and operated by IBM. On information and belief, IBM also sells or leases to customers its Cloud Object Storage System software for operation on the customers' own computer hardware (*e.g.*, servers) and/or computers configured with Cloud Object Storage System software for operation by the customers.[28] On information and belief, IBM also provides installation support and/or software maintenance support in conjunction with these leases or sales.

[26] *See, e.g.*, IBM Red Paper at 19.

[27] IBM Definitive Guide at 5.

[28] *IBM Cloud Pricing*, IBM, https://www.ibm.com/cloud/pricing (last accessed Feb. 25, 2025); *see also IBM Cloud Object Storage System*, SHI: GS, https://texas.gs.shi.com/product/36010513/IBM-Cloud-Object-Storage-System (last accessed February 25, 2025); Timothy Prickett Morgan,

E. IBM's Relationship with SFI And Knowledge of the Asserted Patents

50. IBM contracted with SFC to include its technology in IBM products for over a decade. Before May 2019, SFC delivered multiple products to IBM, such as PureApplications Security.

51. SFC and IBM also entered into multiple Joint Development Agreements to develop products, such as: Secure Hadoop Cloud, a security architecture that uses distributed storage and parallel processing; Cloud Gateway, a network protection technology involving the filtering of public and private-facing network traffic; Guardium, a database security tool that protects data and monitors database activity in real-time; and a combination of SFC's SPxCore, SPxBitFiler, and SPxConnect products with IBM's wire speed system.[29]

52. During that time, SFC made several detailed disclosures to IBM of the breadth of SFC's patent portfolio and of its specific patents.

53. In July 2005, SFC first disclosed to IBM the existence of SFC's patent applications. More specifically, on or around July 29, 2005, an SFC representative emailed a SecureParser White Paper and PowerPoint presentation to Christopher Hanson of IBM. The White Paper described SFC's SecureParser as a "unique patent pending cross-platform software module suite that not only incorporates NSA Type 1 cryptographic primitives providing an alternative approach to data security, but also has the ability to provide fault tolerance, increased access speed and physical separation of data. This approach eliminates many of the problems typically associated

IBM Cloud Storage and BRMS Get Subscription Pricing, IT JUNGLE (Jan. 22, 2024), https://www.itjungle.com/2024/01/22/ibm-cloud-storage-and-brms-get-subscription-pricing/.

[29] *IBM and Security First Corp. to Develop Integrated Security Technology*, BUSINESSWIRE (July 29, 2011), https://www.businesswire.com/news/home/20110729005208/en/IBM-and-Security-First-Corp.-to-Develop-Integrated-Security-Technology.

with security solutions including key management and single points of failure." And in the PowerPoint presentation, SFC explained that it had "[n]ine patents . . . in process that provide for broad coverage of the invention."

54. In 2006, SFC informed IBM that another entity called Cleversafe had technology that might infringe SFC's soon-to-be-issued patents. More specifically, on or around October 9, 2006, John Mumaugh of SFC emailed Christopher Hanson of IBM stating that Cleversafe might infringe SFC's future patents. On information and belief, IBM acquired Cleversafe in October 2015. On information and belief, IBM reviewed SFI's patent portfolio in the course of, and/or prior to, its acquisition of Cleversafe, including the Asserted Patents..

55. In 2009, SFC further disclosed its patent portfolio to IBM. In particular, on or around August 24, 2009 John Mumaugh of SFC emailed Christopher Hanson of IBM and others, attaching SFC's published patents. Mr. Mumaugh also disclosed that SFC had several other pending U.S. patents and foreign patents securing its SecureParser technology. Mr. Mumaugh explained that "SFC has patented . . . the whole method and system of [its] technology which of course includes random splitting of data and all associated steps in the SecureParser." And on or around August 31, 2009, John Mumaugh of SFC emailed Ginny Lee of IBM responses to specific questions about SFC's patents and proprietary software. In that email, Mr. Mumaugh explained that "SFC has received several technology patents in support of [its] cryptographic random splitting technology (SecureParser)" and the then-current status of SFC's patent holdings.

56. In 2011, when negotiating a Base Agreement between SFC and IBM, SFC further disclosed its patents to IBM. More specifically, on or around May 2, 2011, John Mumaugh of SFC emailed Todd Cushman and Christopher Hanson of IBM, providing information about SFC's patent portfolio at that time. Among other things, Mr. Mumaugh included a presentation stating

that "SFC['s] IP Portfolio is extensive" and that SFC had a "[c]ombination of 100 awarded patents and pending patents that span mainly the United States, Australia, Brazil, Canada, China, the European Community, and Hong Kong."

57. Then, in November 2012, SFC and IBM entered into the Base Agreement, pursuant to which SFC agreed to grant IBM, "upon issuance of a [Statement of Work], a nonexclusive, worldwide, perpetual and paid-up license under any patents and patent applications licensable by [SFC] related to the Products and Deliverables to make, have made only for and on behalf of [IBM], use, import, export, sell and otherwise transfer those Products and Deliverables for which a software or copyright license or other right or permission is granted under such [Statement of Work]. There are no other patent licenses granted expressly or impliedly."

58. In 2013, SFC further disclosed the breadth of its patent portfolio. In particular, on or around March 12, 2013, John Mumaugh of SFC emailed Christopher Hanson of IBM that SFC had obtained "overwhelming patent registration" with over "150 patents granted" and that "[i]t will be difficult for companies to compete without licensing or risk of penalty." By this point, the '802 Patent had already issued and was publicly available.

59. In April 2015, SFC sent IBM approximately three gigabytes of information about SFC's technology and intellectual property, including information regarding the technology and intellectual property related to the Asserted Patents. By this point, the '194 Patent had already issued and was publicly available.

60. And in 2016, SFC again disclosed the details of its patent portfolio. On or around November 30, 2016, John Mumaugh of SFC emailed Michael Loria of IBM a summary of its patent portfolio that included 59 U.S. patents and 86 foreign patents. By this point, all of the Asserted Patents had issued and were publicly available.

61. On multiple occasions during SFC's close working relationship with IBM—but, especially between approximately 2009 and 2019—SFC and IBM had many interactions and meetings about, among other things, the methods and technology underlying the Asserted Patents. On information and belief, IBM repeatedly expressed to SFC that IBM wanted to incorporate this technology into its product offerings, that doing so would result in substantial royalties to SFC, and that the technology was enormously valuable to IBM.

62. By May 2019, IBM had cancelled its last contract with SFC. However, IBM continued to use SFC's technologies without permission or compensation to SFC.

63. As a result of SFC's close working relationship with IBM, including its various Joint Development Agreements, SFC's patent-related disclosures, and IBM's continued use of SFC technologies after cancellation of its contractual relationship with SFC, IBM had actual knowledge of, or was at least willfully blind to (1) the existence of all of the Asserted Patents more than six years before the initiation of this litigation and (2) its infringement of the Asserted Patents.

FIRST CLAIM FOR RELIEF

(Infringement of U.S. Patent No. 9,135,456)

64. SFI repeats, re-alleges, and incorporates herein by reference the allegations of Paragraphs 1-63 of its Complaint.

65. On September 15, 2015, United States Patent Number 9,135,456, entitled "Secure Data Parser Method and System," was duly and legally issued to inventors Mark S. O'Hare, Rick L. Orsini, Roger S. Davenport, and Steven Winick. A true and correct copy of the '456 Patent is attached to this Complaint as Exhibit A.

66. SFI is the owner by assignment of the entire right, title, and interest in and to the '456 Patent, including the right to seek damages and any remedies for past, current, and future infringement thereof.

67. The ’456 Patent claims priority to United States Provisional Patent Application No. 60/622,146 filed on October 25, 2004.

68. The ’456 Patent is valid and enforceable.

69. SFI is informed and believes, and on that basis alleges, that IBM has infringed and is currently infringing one or more claims of the ’456 Patent, in violation of 35 U.S.C. § 271 *et seq.*

70. IBM infringes literally and/or under the doctrine of equivalents, in violation of 35 U.S.C. § 271(a) by, among other things, making, using, offering to sell, and/or selling within this District and elsewhere in the United States, without authority or license, IBM products and services that fall within the scope of one or more claims of the ’456 Patent.

71. For example, as discussed below, the methods and products with which IBM performs its encryption technology infringes at least Claim 1 of the ’456 Patent.

Claim 1[pre]: A method performed by a computer system for securing a first data set while providing recoverability of the first data set in the event of unavailability of data at one or more storage locations, the method comprising:

72. IBM meets the preamble, regardless of whether the preamble is found to be limiting. For example, IBM’s Cloud Object Storage System performs a method for securing a first data set while providing recoverability of the first data set in the event of unavailability of data at one or more storage locations.

73. IBM describes its Cloud Object Storage System as “us[ing] an innovative approach for cost-effectively storing large volumes of unstructured data while still ensuring security, availability, and reliability.”[30] And “[a]t the foundation of the Cloud Object Storage System is a

[30] IBM Definitive Guide at 3.

technology that is called information dispersal. Information dispersal is the practice of using erasure codes to create redundancy for transferring and storing data."[31]

74. As described above, IBM Cloud Object Storage System uses "Information Dispersal Algorithms (IDA) to separate data into unrecognizable 'slices'" and those "slices" are distributed "across a network of data centers, making transmission and storage of data inherently private and secure."[32]

Claim 1[a]: reading the first data set from a memory that stores the first data set,

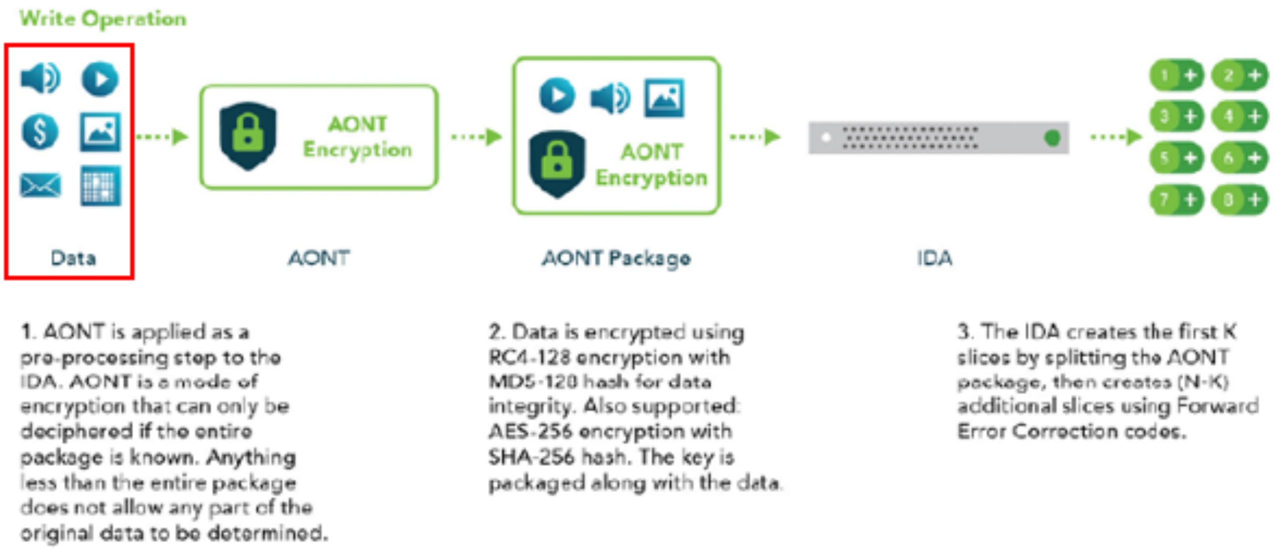
75. IBM meets this limitation. For example, IBM's Cloud Object Storage System reads the first data set from memory.

76. As IBM explains, and as shown in the exemplary IBM graphic below, the Cloud Object Storage System "uses three steps for slicing, dispersing, and retrieving data," with the first step being "[d]ata is virtualized, transformed, sliced, and dispersed using IDAs," as shown in the graphic below:[33] In order to perform these steps on a first data set, the System is required to read that data set from memory.

[31] IBM White Paper at 3.

[32] *Data Security*, IBM, https://cloud.ibm.com/docs/cloud-object-storage?topic=cloud-object-storage-security (last updated April 17, 2024).

[33] IBM White Paper at 9 (red annotation added).



1. AONT is applied as a pre-processing step to the IDA. AONT is a mode of encryption that can only be deciphered if the entire package is known. Anything less than the entire package does not allow any part of the original data to be determined.

2. Data is encrypted using RC4-128 encryption with MD5-128 hash for data integrity. Also supported: AES-256 encryption with SHA-256 hash. The key is packaged along with the data.

3. The IDA creates the first K slices by splitting the AONT package, then creates (N-K) additional slices using Forward Error Correction codes.

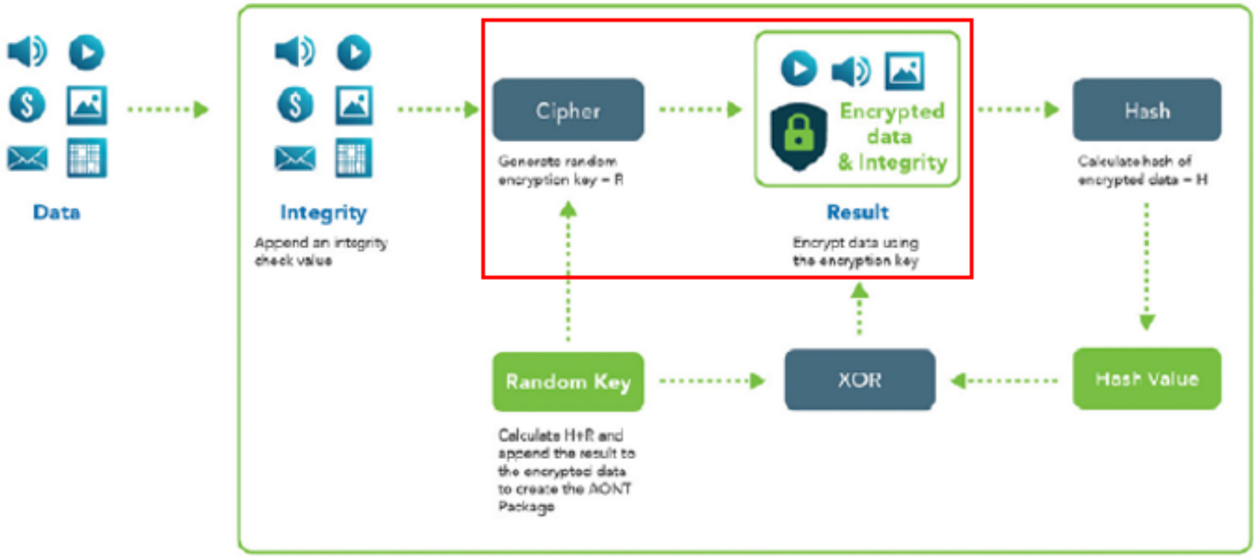
<u>Claim 1[b] processing the first data set to produce a second data set, the processing comprising:</u>

77. IBM meets this limitation. For example, IBM's Cloud Storage processes the first data set to produce a second data set.

78. As IBM explains "IBM SecureSlide™ technology is used to help ensure confidentiality, integrity and availability of data stored on a Cloud Object Storage System. SecureSlice combines two algorithms: an IDA [Information Dispersal Algorithm] and an All-or-Nothing Transform (AONT)."[34] As described below in connection with limitations 1[b][i]-1[b][iii], the IBM Cloud Object Storage System combines the encrypted data and a "resultant" comprising the logical combination of the key used to encrypt the data and a hash of the encrypted data into a single package, which corresponds to the second data set recited in the claim.[35]

[34] IBM White Paper at 9.

[35] IBM Red Paper at 17-18; IBM White Paper at 10 (red annotation added).


Data
Integrity
Append an integrity check value
Cipher
Generate random encryption key – R
Encrypted data & Integrity
Result
Encrypt data using the encryption key
Hash
Calculate hash of encrypted data – H
Random Key
Calculate H+R and append the result to the encrypted data to create the AONT Package
XOR
Hash Value

<u>Claim 1[b][i] encrypting the first data set using an encryption key to produce an encrypted data set, and</u>

79. IBM meets this limitation. For example, IBM's Cloud Object Storage System encrypts the first data set using an encryption key to produce an encrypted data set.

80. For example, as IBM explains, "SecureSlice uses an *all-or-nothing-transform* (AONT) to encrypt the data."[36] And this process involves "[g]enerat[ing] a random encryption key" and "[e]ncrypt[ing] data by using encryption key."[37]

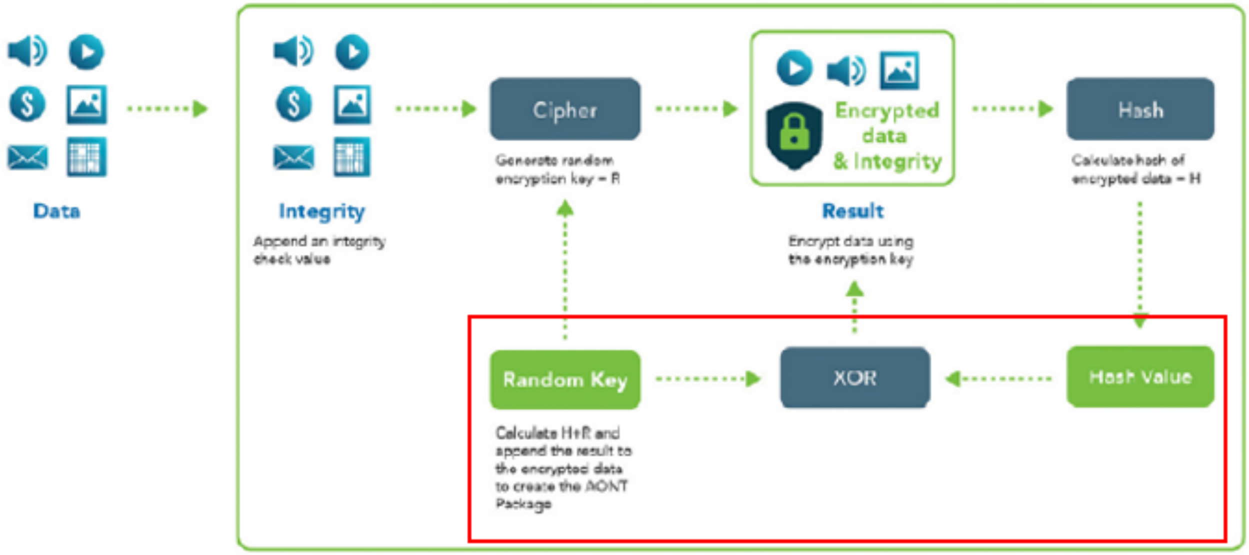
<u>Claim 1[b][ii] logically combining at least a portion of the encrypted data set with the encryption key to produce a resultant,</u>

81. IBM meets this limitation literally or through the doctrine of equivalents. For example, IBM's Cloud Object Storage System logically combines at least a portion of the encrypted data set with the encryption key to produce a resultant.

[36] IBM Red Paper at 17.

[37] *Id.*

82. For example, IBM explains that the "IBM COS SecureSlice AONT encryption process" involves "[c]alculat[ing] the hash of encrypted data," which comprises at least a portion of the encrypted data set or its equivalent, and then "[c]alculat[ing] the exclusive-OR (XOR) of hash and encryption key."[38] The "result" of this logical combination is the resultant required by the claim or its equivalent. This process is shown in the exemplary IBM graphic below[39]:


Data
Integrity
Append an integrity check value
Cipher
Generate random encryption key – R
Encrypted data & Integrity
Result
Encrypt data using the encryption key
Hash
Calculate hash of encrypted data – H
Random Key
Calculate H+R and append the result to the encrypted data to create the AONT Package
XOR
Hash Value

83. To the extent this limitation is not literally present, the differences between the claim limitation and IBM's Cloud Object Storage System are insubstantial, and IBM's Cloud Object Storage System performs substantially the same function, in substantially the same way, to achieve the same result. Specifically, IBM's Cloud Object Storage System performs substantially the same function (creating a resultant based on the encryption key and data related to the first data set), in substantially the same way (by using the XOR logical operator against a hash of the encrypted data and the encryption key), to achieve the same result (storing with the encrypted data

[38] IBM Red Paper at 17.

[39] IBM White Paper at 10 (red annotations added).

an obfuscated version of the encryption key which, along with the encrypted data, can be used to retrieve the encryption key).

Claim 1[b][iii] wherein the second data set comprises the encrypted data set and the resultant, and wherein all of the at least a portion of the encrypted data set is required to recover the encryption key based on the resultant;

84. IBM meets this limitation either directly or through the doctrine of equivalents. For example, within IBM's Cloud Object Storage System, the second data set comprises the encrypted data set and the resultant, wherein all of the "at least a portion" of the encrypted data set or its equivalent is required to recover the encryption key based on the resultant.

85. As explained above, IBM's Cloud Object Storage System creates a resultant, or its equivalent, when it calculates the "exclusive-OR (XOR) of hash and encryption key" to form a "result."[40] It creates the required second data set when it "[a]ppend[s] the result to the encrypted data to create the AONT package."[41] At least a portion of the encrypted data set, or its equivalent, is required to recover the encrypted key because, in IBM Cloud Object Storage the encryption key has been XORed against the hash of the encrypted data set, and the key can only be recovered by carrying out the reverse operation using the resultant and the hash of the data set.

86. To the extent this limitation is not literally present, the differences between the claim limitation and IBM's Cloud Object Storage System are insubstantial, and IBM's Cloud Object Storage System performs substantially the same function, in substantially the same way, to achieve the same result. Specifically, IBM's Cloud Object Storage System performs substantially the same function (creating a resultant based on the encryption key and data related to the first data set), in substantially the same way (by using the XOR logical operator against a hash of the

[40] IBM Red Paper at 17.

[41] *Id.*

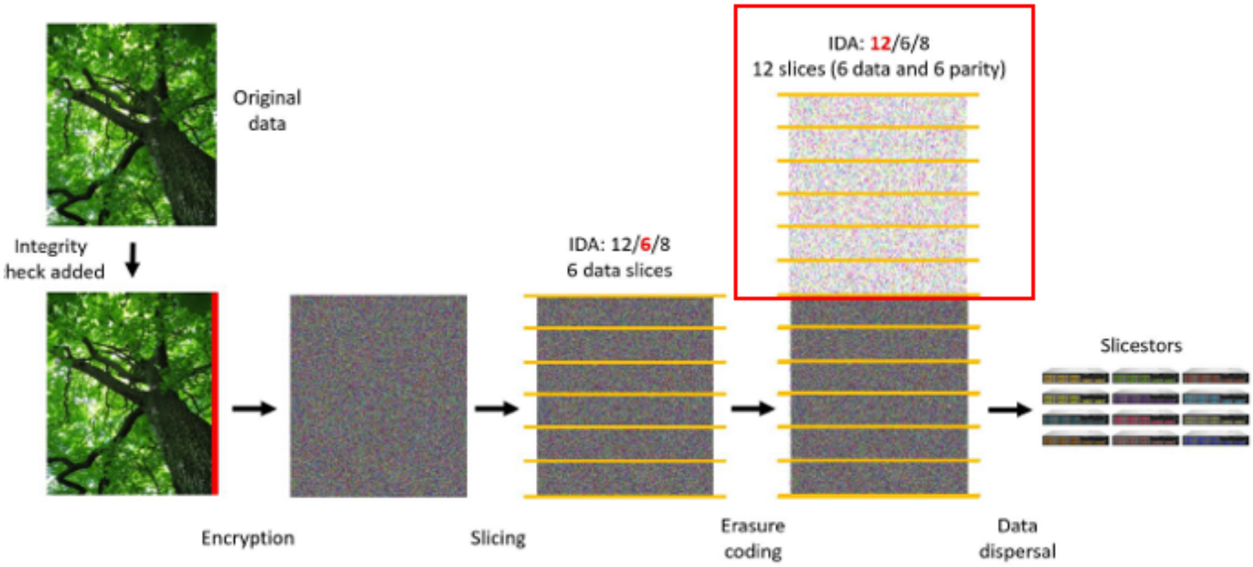
encrypted data and the encryption key), to achieve the same result (storing with the encrypted data an obfuscated version of the encryption key which, along with the encrypted data, can be used to retrieve the encryption key).

Claim 1[c] producing redundancy information based on information in the second data set;

87. IBM meets this limitation. For example, IBM's Cloud Object Storage System produces redundancy information based on information in the second data set.

88. More specifically, IBM's Cloud Object Storage System produces redundancy information by using IDAs (*i.e.*, Information Dispersal Algorithms), which "use[] erasure code as a means to create redundancy for transferring and storing data."[42] As IBM explains, "[w]ith dispersed storage, only a subset of slices is needed to retrieve the data."[43]

89. The redundancy is shown in the exemplary IBM graphic below:[44]



[42] IBM Product Guide at 4.

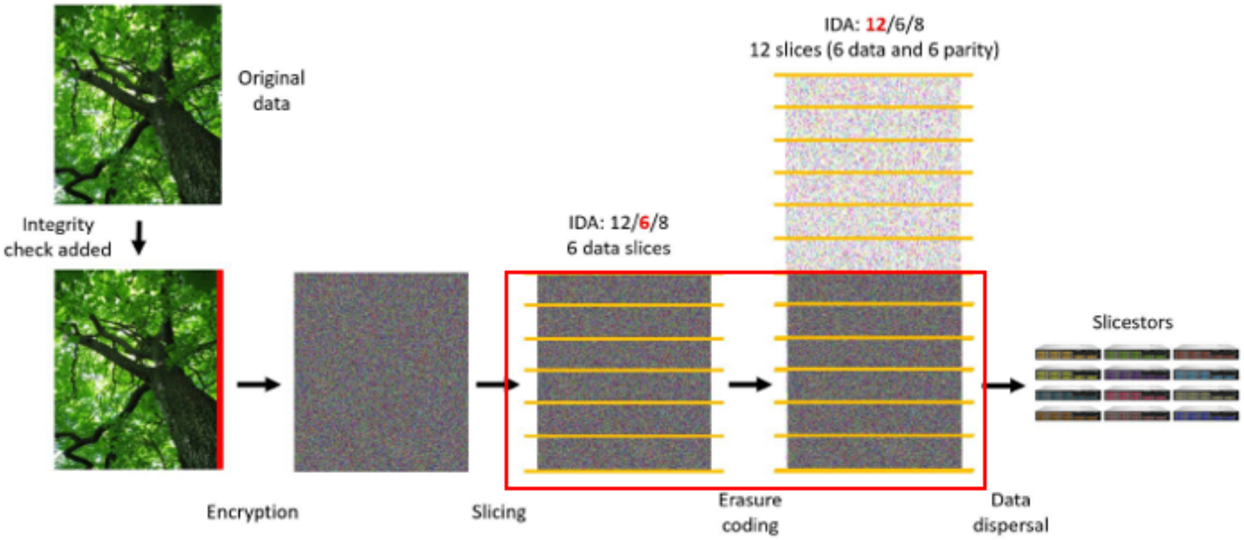
[43] *Id.*

[44] IBM Red Paper at 19 (red annotation added).

<u>Claim 1[d] producing a plurality of n shares from the second data set, wherein each of the n shares comprises at least some of the second data set; and</u>

90. IBM meets this limitation. For example, IBM's Cloud Object Storage System produces a plurality of n shares from the data set, wherein each of the n shares comprises at least some of the second data set.

91. As IBM explains, "[t]he IBM COS erasure coding process includes the following steps: 1. The AONT package is sliced to [a] read threshold number of slices. 2. Erasure coding creates encoded slices so that the total number of slices (data slices plus encoded slices) is equal to IDA width."[45] This is shown on the exemplary IBM graphic below.[46] The read threshold number of slices corresponds to the n shares required by the claim.



[45] IBM Red Paper at 18.

[46] *Id.* at 19 (red annotations added).

Claim 1[e] storing the plurality of n shares and the redundancy information, wherein at least some of the n shares are stored in separate storage locations and the first data set is recoverable using at least a threshold number, less than n, of the plurality of n shares.

92. IBM meets this limitation. For example, IBM's Cloud Object Storage System stores the plurality of n shares and the redundancy information, wherein at least some of the n shares are stored in separate storage locations and the first data set is recoverable using at least a threshold number, less than n, of the plurality of n shares.

93. IBM explains that "[a]fter erasure coding, data is distributed to the Slicestor nodes."[47] And "[i]f the storage pool is configured to use SD Mode, each Slicestor node in a device set stores a single slice."[48] But "[i]f the storage pool is configured to use CD Mode, each Slicestor node stores multiple slices. The system ensures that the slices are not stored on the same drives within the same chassis."[49] Each drive within a chassis in a Slicestor node is a separate storage location.

94. IBM also explains that after "[s]lices are distributed to some combination of separate disks, storage nodes, and geographic locations," "[t]he data is retrieved from a subset of slices," which is less than the number of slices that were created.[50]

* * *

95. To the extent any limitation of claim 1 of the '456 Patent is not met literally, on information and belief, each such limitation is met through application of the doctrine of equivalents. The elements of the Accused Product are at most insubstantially different from, and

[47] *Id.* at 19.

[48] *Id.*

[49] *Id.* at 20.

[50] IBM Definitive Guide at 5-6.

perform substantially the same function in substantially the same way to achieve the same result as, each limitation of claim 1 of the ’456 Patent.

96. On information and belief, IBM had knowledge of, or was willfully blind to, the claims of the ’456 Patent. On information and belief, IBM had knowledge of, or was willfully blind to, the fact that its conduct constituted induced, or contributed to infringement of the ’456 Patent.

97. As a result of IBM’s infringement of the ’456 Patent, SFI has been damaged. SFI is entitled to recover from IBM damages sustained as a result of IBM’s wrongful acts sufficient to compensate SFI for the infringement in an amount subject to proof at trial, and in no event less than a reasonable royalty.

98. To the extent 35 U.S.C. § 287 is determined to be applicable, on information and belief, its requirements have been satisfied with respect to the ’456 Patent.

99. IBM also indirectly infringes one or more claims of the ’456 Patent in violation of 35 U.S.C. § 271(b) by actively inducing others (including, but not limited to, IBM’s customers) to infringe the ’456 Patent, by, among other things, providing, on information and belief, instructions, manuals, technical assistance, and promotional materials relating to the installation, use, operation, and maintenance of the IBM Cloud Object Storage system in the United States. Defendants’ inducement is ongoing.

100. IBM also indirectly infringes one or more claims of the ’456 Patent in violation of U.S.C. § 271(c) by contributing to others’ infringement of the ’456 Patent by, on information and belief, selling, offering to sell, and importing components that its customers use to build and assemble, and that its customers use to operate, the IBM Cloud Object Storage system, which components constitute a material part of the claimed invention of the ’456 Patent. IBM knows

that the components of the IBM Cloud Object Storage are especially made or adapted for use in infringement of the ’456 Patent, and those components are not a staple article or commodity of commerce suitable for substantial non-infringing use. Defendants’ contributory infringement is ongoing.

101. SFI has suffered and continues to suffer irreparable injury as a direct and proximate result of IBM’s infringement for which there is no adequate remedy at law. Unless IBM is enjoined, SFI will continue to suffer such irreparable injury.

SECOND CLAIM FOR RELIEF

(Infringement of U.S. Patent No. 8,904,194)

102. SFI repeats, re-alleges, and incorporates herein by reference the allegations of Paragraphs 1-101 of its Complaint.

103. On December 2, 2014, United States Patent Number 8,904,194, entitled “Secure Data Parser Method and System,” was duly and legally issued to inventors Rick L. Orsini, Mark S. O’Hare, Roger S. Davenport, and Steven Winick. A true and correct copy of the ’194 Patent is attached to this Complaint as Exhibit B.

104. SFI is the owner by assignment of the entire right, title, and interest in and to the ’194 Patent, including the right to seek damages and any remedies for past, current, and future infringement thereof.

105. The ’194 Patent claims priority to United States Provisional Patent Application No. 60/622,146 filed on October 25, 2004.

106. The ’194 Patent is valid and enforceable.

107. SFI is informed and believes, and on that basis alleges, that IBM has infringed and is currently infringing one or more claims of the ’194 Patent, in violation of 35 U.S.C. § 271 *et seq.*

108. IBM infringes literally and/or under the doctrine of equivalents, in violation of 35 U.S.C. § 271(a) by, among other things, making, using, offering to sell, and/or selling within this District and elsewhere in the United States, without authority or license, IBM products and services that fall within the scope of one or more claims of the '194 Patent.

109. For example, as discussed below, the methods and products with which IBM performs its encryption technology infringes at least Claim 1 of the '194 Patent.

Claim 1[pre] A method for securely storing and retrieving data, the method comprising:

110. IBM meets the preamble, regardless of whether the preamble is limiting. For example, IBM's Cloud Object Storage System performs a method for securely storing and retrieving data.

111. IBM describes its Cloud Object Storage System as "us[ing] an innovative approach for cost-effectively storing large volumes of unstructured data while helping ensure security, availability, and reliability."[51] And "[w]ith Cloud Object Storage dispersed storage technology, transmission and storage of data are inherently private and secure. No complete copy of the data resides in any single storage node, and only a subset of nodes needs to be available to fully retrieve the data on the network."[52]

Claim 1[a] generating, using an electronic computing system that includes processing circuitry, a plurality of shares by performing a cryptographic operation on a data set and distributing the data set in the plurality of shares such that the data set can be reconstructed using any subset of the shares that includes at least a minimum number less than all of shares;

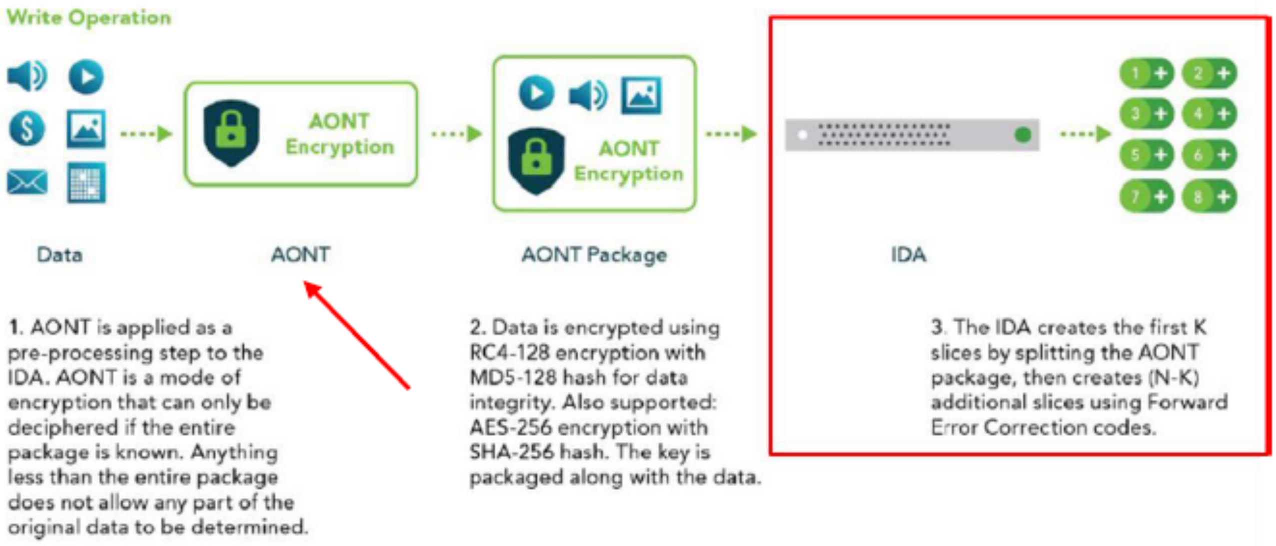
112. IBM meets this limitation. For example, IBM's Cloud Object Storage generates, using an electronic computing system that includes processing circuitry, a plurality of shares by

[51] IBM White Paper at 2.

[52] *Id.*

performing a cryptographic operation on a data set and distributing the data set in the plurality of shares such that the data set can be reconstructed using any subset of the shares that includes at least a minimum number less than all of shares.

113. IBM explains that "[a]t a basic level, the Cloud Object Storage System uses three steps for slicing, dispersing, and retrieving data. 1. Data is virtualized, transformed, sliced, and dispersed by using IDAs. . . . 2. Slices are distributed to separate disks, storage nodes, and geographic locations. . . . 3. The data is retrieved from a subset of slices."[53] The data is thus "transformed" using AONT, is a cryptographic operation performed on the data, and distributed (using IDAs) into "slices" corresponding to the "shares" recited in the claim. These features are shown on the exemplary IBM graphic below:[54]



1. AONT is applied as a pre-processing step to the IDA. AONT is a mode of encryption that can only be deciphered if the entire package is known. Anything less than the entire package does not allow any part of the original data to be determined.

2. Data is encrypted using RC4-128 encryption with MD5-128 hash for data integrity. Also supported: AES-256 encryption with SHA-256 hash. The key is packaged along with the data.

3. The IDA creates the first K slices by splitting the AONT package, then creates (N-K) additional slices using Forward Error Correction codes.

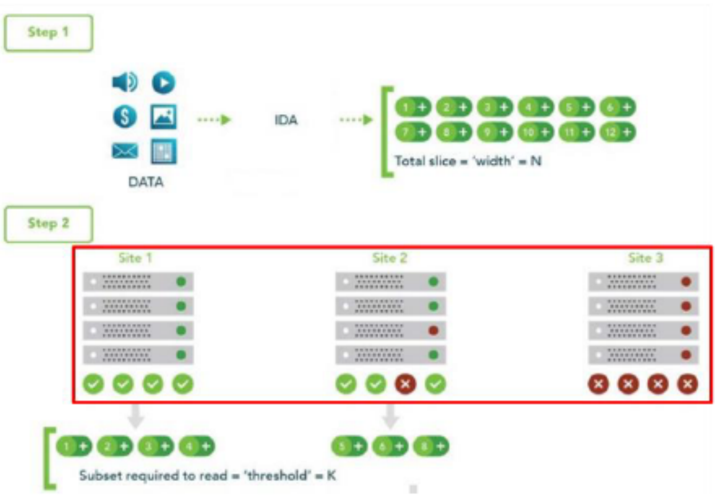
<u>**Claim 1[b] storing the plurality of shares at a plurality of storage devices;**</u>

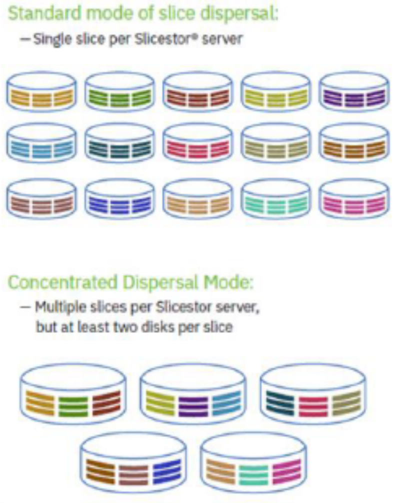
114. IBM meets this limitation. For example, IBM's Cloud Object Storage System stores the plurality of shares at a plurality of storage devices.

[53] *Id.* at 4.

[54] *Id.* at 9 (red annotations added).

115. IBM explains that "[a]fter erasure coding, data is distributed to the Slicestor nodes."[55] And "[i]f the storage pool is configured to use SD Mode, each Slicestor node in a device set stores a single slice."[56] But "[i]f the storage pool is configured to use CD Mode, each Slicestor node stores multiple slices. The system ensures that the slices are not stored on the same drives within the same chassis."[57] The CD Mode "allocates multiple slices of an object to the same storage node (but no more than one slice per disk) to optimize the reliability, availability and efficiency of the Cloud Object Storage system."[58] This is shown on the exemplary IBM graphic below:[59]




Standard mode of slice dispersal:
– Single slice per Slicestor® server
Concentrated Dispersal Mode:
– Multiple slices per Slicestor server,
but at least two disks per slice

[55] IBM Red Paper at 19.

[56] *Id.*

[57] *Id.* at 20.

[58] IBM White Paper at 15.

[59] *Id.*; IBM Definitive Guide at 5 (showing steps 1 and 2; red annotation added).

<u>Claim 1[c] receiving, at the electronic computing system, request to retrieve the data set;</u>

116. IBM meets this limitation. For example, IBM's Cloud Object Storage System receives, at the electronic computing system, requests to retrieve the data set.

117. IBM provides that "[o]ne of the advantages of IBM COS is data can be read by using any Accesser node within the system, even from ones on different sites."[60] The "read operations" include a number of steps, including:

> The client application issues a read request that is sent to one of the Accesser nodes. 2. The Accesser node instructs the Slicestor nodes to send the required data slices. 3. After the read threshold number of slices is received, the Accesser node decodes the object and verifies its integrity. 4. The reconstructed data is sent to the client application by using the S3 protocol.[61]

118. The "read request" therefore corresponds to a request to retrieve the data.

<u>Claim 1[d] identifying from the plurality of storage devices a set of fastest-responding storage devices necessary to retrieve the minimum number of shares, wherein the set of fastest-responding storage devices are identified based at least in part on the response time of the storage devices;</u>

119. IBM meets this limitation. For example, IBM's Cloud Object Storage System identifies from the plurality of storage devices a set of fastest-responding storage devices necessary to retrieve the minimum number of shares, wherein the set of fastest-responding storage devices are identified based at least in part on the response time of the storage devices.

120. According to IBM, in its Cloud Object Storage System, "[d]ata is reassembled in segments, and for each segment, thousands, if not millions, of combinations of slices are examined to determine the best delivery path."[62] IBM further explains that

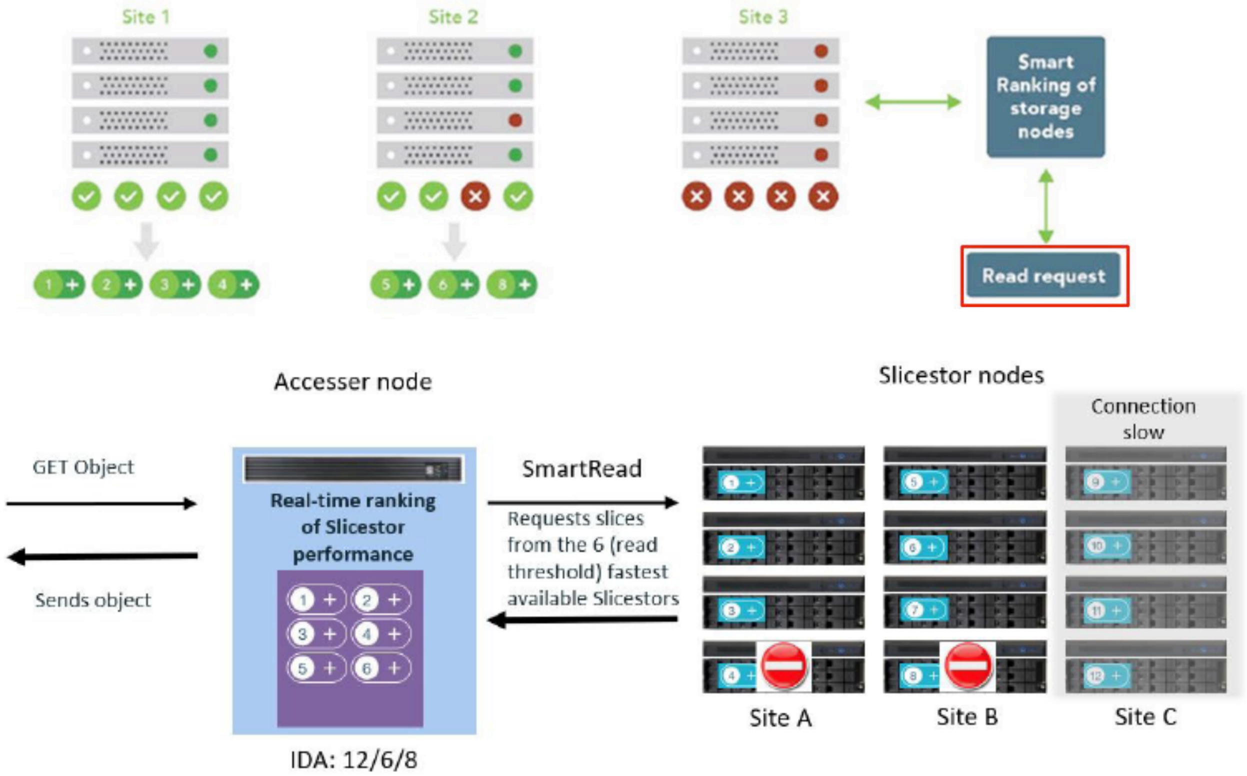
[60] IBM Red Paper at 22.

[61] *Id.*

[62] IBM Definitive Guide at 14.

> SmartRead predicts the optimal network routes and Slicestor nodes from which to most efficiently retrieve data. ***SmartRead ranks Slicestor nodes by real-time performance and requests the optimal combination (read threshold number) of slices to re-create the data***. If a slice request is not performing, SmartRead requests the missing slice from another node. This feature is always on and significantly increases overall system read performance in cases when, for example, the network connection to some Slicestor nodes are slower or a limping Slicestor node in the system.[63]

121. Below is an exemplary IBM graphic showing how this process works.[64]



122. On information and belief, "real-time performance" is based, at least in part, on the response time of the Slicestor nodes, including because SmartRead "[r]equests slices from the . . . fastest available Slicestors."[65]

[63] IBM Red Paper at 24 (emphasis added).

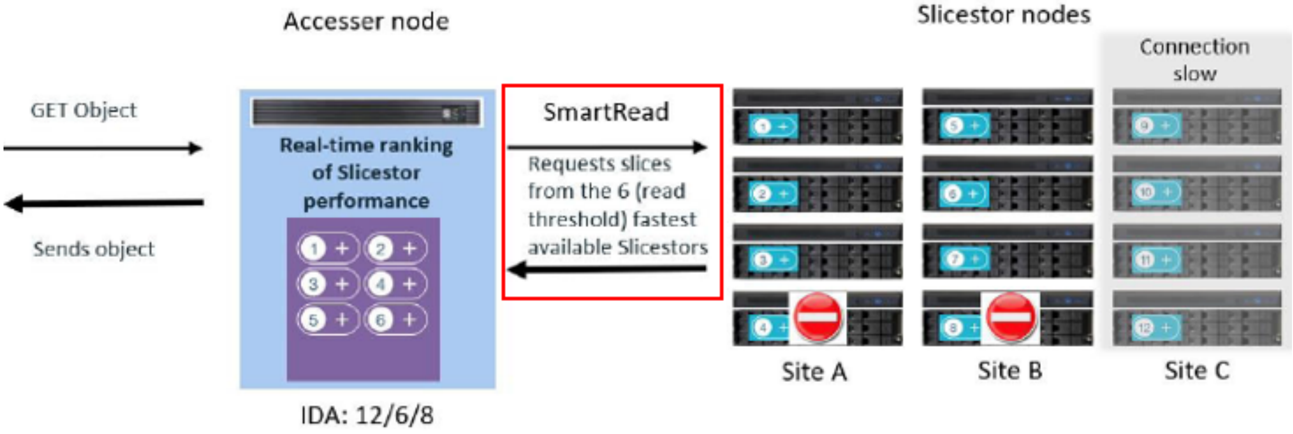
[64] *Id.*

[65] *Id.*

<u>Claim 1[e] retrieving from the set of fastest-responding storage devices, the minimum number of shares;</u>

123. IBM meets this limitation. For example, IBM's Cloud Object Storage retrieves from the set of fastest-responding storage devices, the minimum number of shares.

124. As explained above with respect to Claim 1[d], the SmartRead feature "ranks Slicestor nodes by real-time performance and requests the optimal combination (read threshold number) of slices to re-create the data" and requests the minimum number of shares (*i.e.*, the "read threshold") from the fastest available Slicestors, as shown on the exemplary IBM graphic below.[66]



<u>Claim 1[f] reconstructing the data set using the minimum number of shares; and</u>

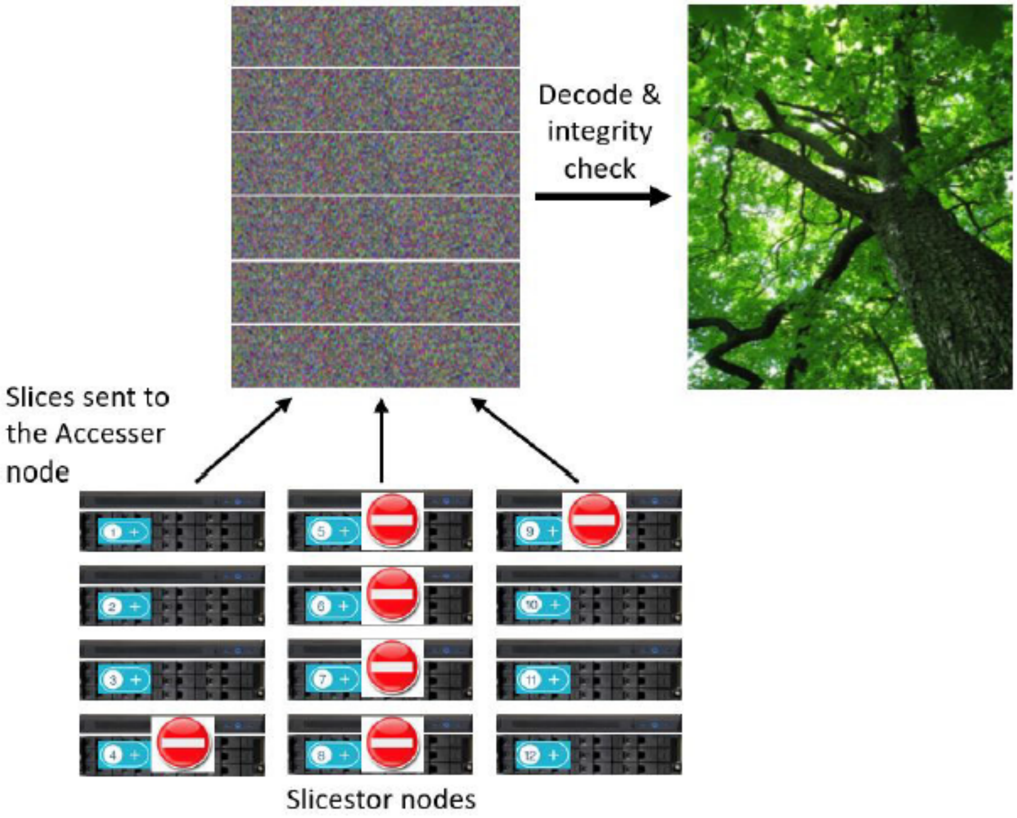
125. IBM meets this limitation. For example, IBM's Cloud Object Storage System reconstructs the data set using the minimum number of shares.

126. IBM explains that, in its system, "[d]ata is reassembled in segments, and for each segment, thousands, if not millions, of combinations of slices are examined to determine the best

[66] IBM Red Paper at 24 (red annotations added).

delivery path."[67] And "[a]fter the read threshold number of slices [(*i.e.*, minimum)] is received, the Accesser node decodes the object and verifies its integrity."[68]

127. IBM explains that data reassembly can be successful "even if multiple Slicestor nodes are down. IBM COS requires only the read threshold number of Slicestor nodes that are available to reconstruct the data."[69] This is shown in the exemplary graphic below.[70]



[67] IBM Definitive Guide at 14.

[68] IBM Red Paper at 22.

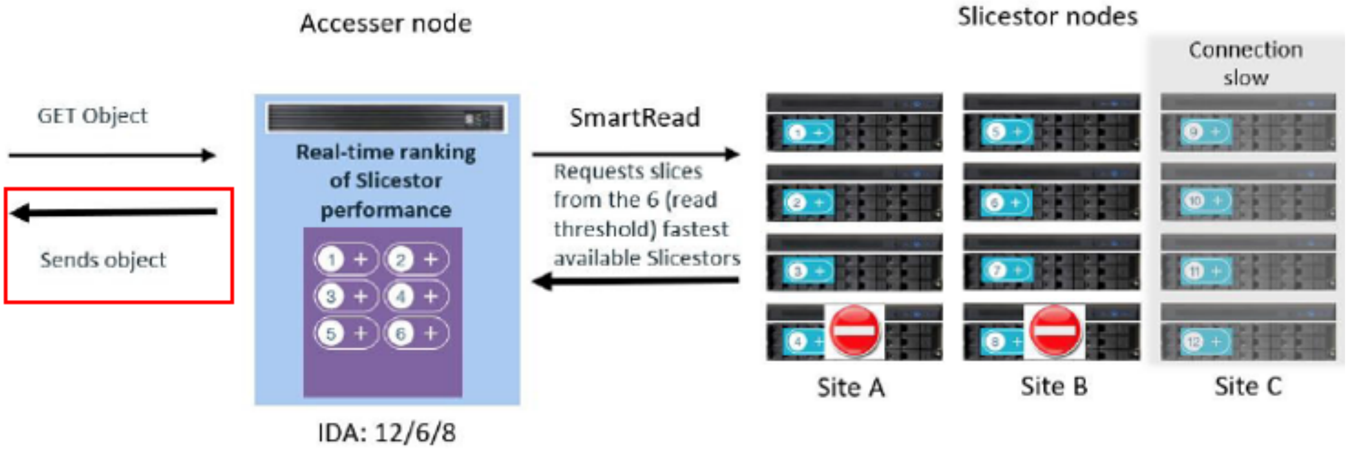
[69] *Id.* at 23.

[70] *Id.*

Claim 1[g] sending the data set responsive to the request.

128. IBM meets this limitation. For example, IBM's Cloud Object Storage System sends the data set responsive to the request.

129. After reconstructing the data, it "is sent to the client application by using the S3 protocol."[71] This is shown on the exemplary IBM graphic below.[72]



* * *

130. To the extent any limitation of claim 1 of the '194 Patent is not met literally, on information and belief, each such limitation is met through application of the doctrine of equivalents. The elements of the Accused Product are at most insubstantially different from, and perform substantially the same function in substantially the same way to achieve the same result as, each limitation of claim 1 of the '194 Patent.

[71] IBM Red Paper at 22.

[72] *Id.* at 24 (red annotation added).

131. On information and belief, IBM had knowledge of, or was willfully blind to, the ’194 Patent. On information and belief, IBM had knowledge of, or was willfully blind to, the fact that its conduct constituted, induced, or contributed to infringement of the ’194 Patent.

132. As a result of IBM’s infringement of the ’194 Patent, SFI has been damaged. SFI is entitled to recover from IBM damages sustained as a result of IBM’s wrongful acts sufficient to compensate SFI for the infringement in an amount subject to proof at trial, and in no event less than a reasonable royalty.

133. To the extent 35 U.S.C. § 287 is determined to be applicable, on information and belief, its requirements have been satisfied with respect to the ’194 Patent.

134. IBM also indirectly infringes one or more claims of the ’194 Patent in violation of 35 U.S.C. § 271(b) by actively inducing others (including, but not limited to, IBM’s customers) to infringe the ’194 Patent, by, among other things, providing, on information and belief, instructions, manuals, technical assistance, and promotional materials relating to the installation, use, operation, and maintenance of the IBM Cloud Object Storage system in the United States. Defendants’ inducement is ongoing.

135. IBM also indirectly infringes one or more claims of the ’194 Patent in violation of U.S.C. § 271(c) by contributing to others’ infringement of the ’194 Patent by, on information and belief, selling, offering to sell, and importing components that its customers use to build and assemble, and that its customers use to operate, the IBM Cloud Object Storage system, which components constitute a material part of the claimed invention of the ’194 Patent. IBM knows that the components of the IBM Cloud Object Storage are especially made or adapted for use in infringement of the ’194 Patent, and those components are not a staple article or commodity of

commerce suitable for substantial non-infringing use. Defendants' contributory infringement is ongoing.

136. SFI has suffered and continues to suffer irreparable injury as a direct and proximate result of IBM's infringement for which there is no adequate remedy at law. Unless IBM is enjoined, SFI will continue to suffer such irreparable injury.

THIRD CLAIM FOR RELIEF

(Infringement of U.S. Patent No. 8,271,802)

137. SFI repeats, re-alleges, and incorporates herein by reference the allegations of Paragraphs 1-136 of its Complaint.

138. On September 18, 2012, United States Patent Number 8,271,802, entitled "Secure Data Parser Method and System," was duly and legally issued to inventors Rick L. Orsini, Mark S. O'Hare, Roger S. Davenport, and Steven Winick. A true and correct copy of the '802 Patent is attached to this Complaint as Exhibit C.

139. SFI is the owner by assignment of the entire right, title, and interest in and to the '802 Patent, including the right to seek damages and any remedies for past, current, and future infringement thereof.

140. The '802 Patent claims priority to United States Provisional Patent Application No. 60/622,146 filed on October 25, 2004.

141. The '802 Patent is valid and enforceable.

142. SFI is informed and believes, and on that basis alleges, that IBM has infringed and is currently infringing one or more claims of the '802 Patent, in violation of 35 U.S.C. § 271 *et seq*.

143. IBM infringes literally and/or under the doctrine of equivalents, in violation of 35 U.S.C. § 271(a) by, among other things, making, using, offering to sell, and/or selling within

this District and elsewhere in the United States, without authority or license, IBM products and services that fall within the scope of one or more claims of the ’802 Patent.

144. For example, as discussed below, the methods and products with which IBM performs its encryption technology infringes at least Claim 1 of the ’802 Patent.

Claim 1[pre] A method for securing a data set, the method steps implemented by a programmed computer system, the method steps comprising:

145. IBM meets the preamble, regardless of whether the preamble is found to be limiting. For example, IBM’s Cloud Object Storage System performs a method for securing a data set, the method steps implemented by a programmed computer system.

146. As explained above, IBM’s Cloud Object Storage System purports to “use[] an innovative approach for cost-effectively storing large volumes of unstructured data while helping ensure security, availability, and reliability.”[73]

Claim 1[a] creating integrity information using the data set;

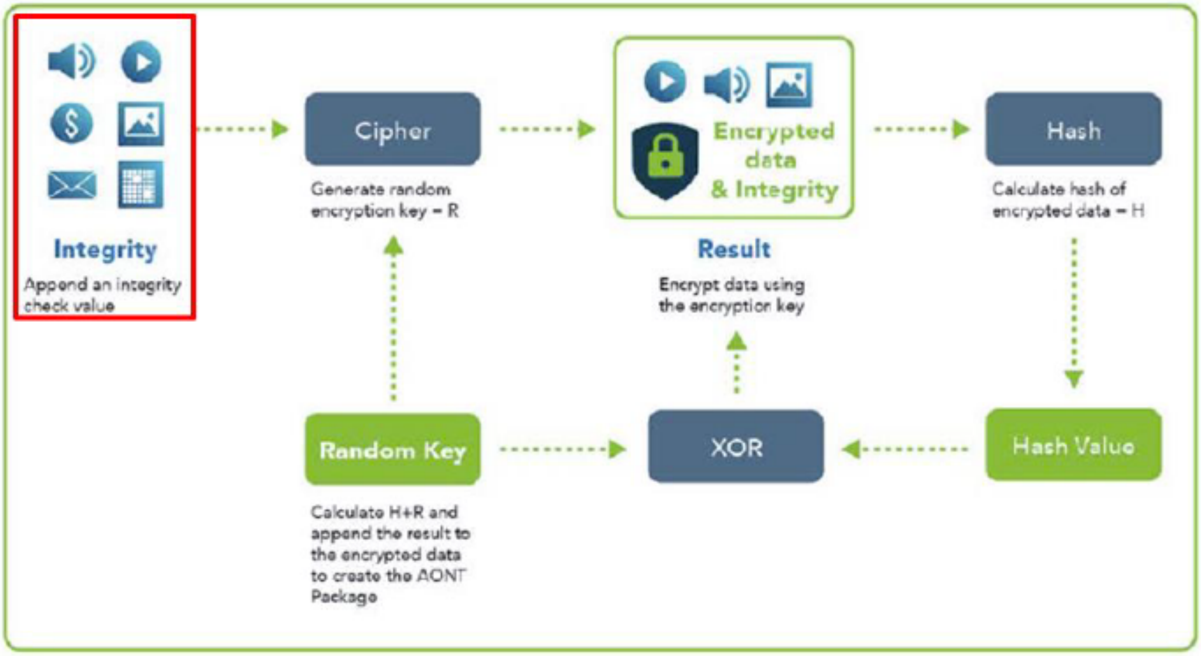
147. IBM meets this limitation. For example, IBM’s Cloud Object Storage System creates integrity information using the data set.

148. As IBM explains, “[w]hen a segment of data is to be stored in a dispersed storage system, an integrity check value is first appended to the data. The integrity check value can be any well-known constant value, if its length is sufficient. This value will be checked after decoding to ensure that no corruption occurred.”[74] The “IBM COS SecureSlice AONT encryption process”

[73] IBM White Paper at 2.

[74] *Id.* at 10.

involves "[a]ppend[ing] an integrity check value to the segment."[75] This can be seen on the exemplary IBM graphic below.[76]


Integrity
Append an integrity check value
Cipher
Generate random encryption key – R
Encrypted data & Integrity
Result
Encrypt data using the encryption key
Hash
Calculate hash of encrypted data – H
Random Key
Calculate H+R and append the result to the encrypted data to create the AONT Package
XOR
Hash Value

Claim 1[b] encrypting the data set based on an encryption key to produce an encrypted data set;

149. IBM meets this limitation. For example, IBM's Cloud Object Storage System encrypts the data set based on an encryption key to produce an encrypted data set.

150. As explained above, IBM's "SecureSlice uses an all-or-nothing-transform (AONT) to encrypt the data. AONT is a type of encryption in which the information can be deciphered only if all the content is known."[77] "[T]he IBM COS SecureSlice AONT encryption process" involves "[g]enerat[ing] [a] random encryption key" and "[e]ncrypt[ing] data by using encryption key."[78]

[75] IBM Red Paper at 17.

[76] IBM White Paper at 10 (red annotations added).

[77] IBM Red Paper at 17.

[78] *Id.*

Claim 1[c] generating data splitting information, wherein the data splitting information is usable to determine into which of a plurality of shares of data a unit of data of the encrypted data set will be placed;

151. IBM meets this limitation directly or through the doctrine of equivalents. For example, IBM's Cloud Object Storage System generates data splitting information, wherein the data splitting information is usable to determine into which of a plurality of shares of data a unit of data of the encrypted data set will be placed.

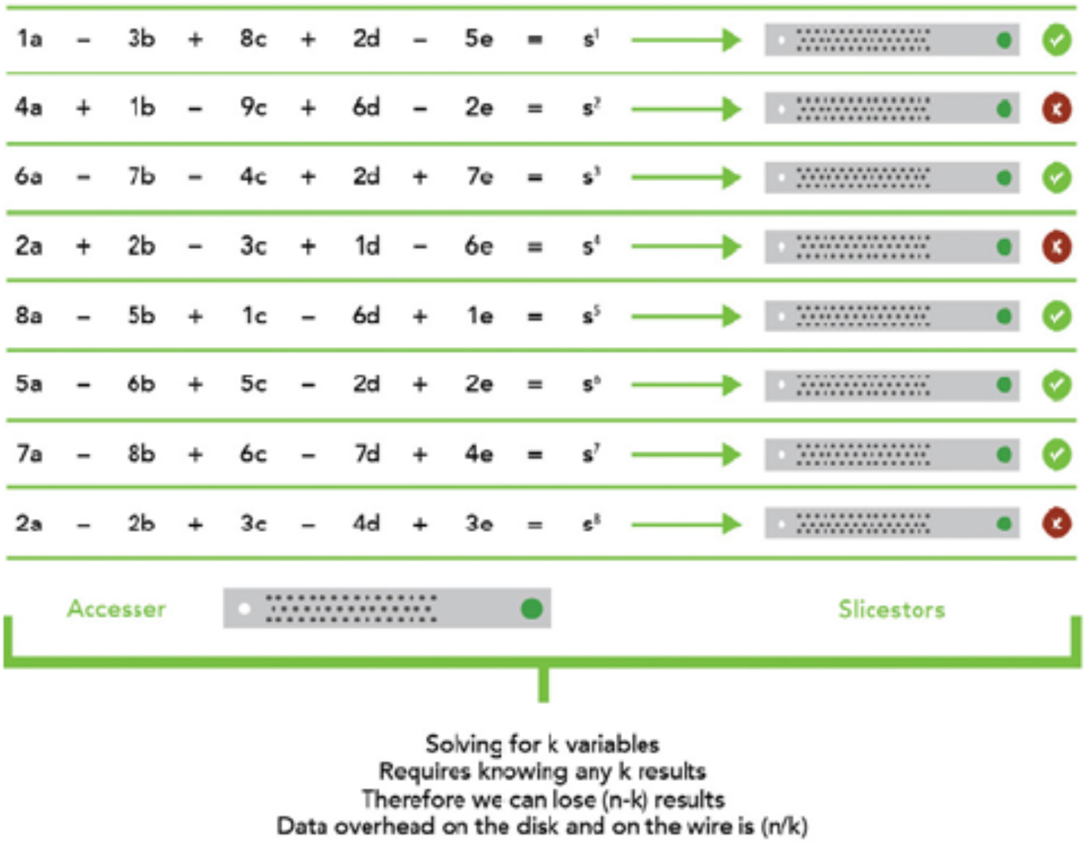
152. IBM documents explain that the information dispersal technology is "[a]t the foundation of the Cloud Object Storage System."[79] This technology creates the "data splitting information" required by the claim or its equivalent.

153. IBM explains how the IDAs work as follows: "How do these IDAs work? In algebra, when you have a system of equations with five variables, you can solve for those variables when you have at least five outputs from different equations using those variables."[80] The exemplary figure below "shows five variables (a through e) and eight different equations that use these variables, with each yielding a different output. To understand how information dispersal works, imagine the five variables are bytes. Following the eight equations, we can compute eight results, each of which is a byte. To solve for the original five bytes, we may use any five of the resulting eight bytes."[81] The equations described by IBM correspond to the "data splitting information" recited by the claim.

[79] IBM White Paper at 3.

[80] *Id.* at 3.

[81] *Id.*


1a – 3b + 8c + 2d – 5e = s^1
4a + 1b – 9c + 6d – 2e = s^2
6a – 7b – 4c + 2d + 7e = s^3
2a + 2b – 3c + 1d – 6e = s^4
8a – 5b + 1c – 6d + 1e = s^5
5a – 6b + 5c – 2d + 2e = s^6
7a – 8b + 6c – 7d + 4e = s^7
2a – 2b + 3c – 4d + 3e = s^8
Accesser
Slicestors
Solving for k variables
Requires knowing any k results
Therefore we can lose (n-k) results
Data overhead on the disk and on the wire is (n/k)

154. To the extent this limitation is not literally present, the differences between the claim limitation and IBM's Cloud Object Storage are insubstantial, and IBM's Cloud Object Storage System performs substantially the same function, in substantially the same way, to achieve the same result. Specifically, IBM's Cloud Object Storage performs substantially the same function (placing units of data from the encrypted dataset into the shares), in substantially the same way (using an algorithm), to achieve the same result (to assign each unit of data into one or more of a plurality of shares of data).

Claim 1[d] separating the encrypted data set into the plurality of shares based on the data splitting information;

155. IBM meets this limitation. For example, IBM's Cloud Object Storage System separates the encrypted data set into the plurality of shares based on the data splitting information.

156. As explained above, IBM's IDAs "split data into inherently secure slices."[82] These are the "shares" of the claim limitation.

<u>Claim 1[e] including in the plurality of shares data indicative of (a) the encryption key and (b) the integrity information; and</u>

157. IBM meets this limitation. For example, IBM's Cloud Object Storage includes in the plurality of shares data indicative of (a) the encryption key and (b) the integrity information.

158. In IBM's system, data indicative of the encryption key and integrity information are appended to and packaged with the encrypted data, so they are necessarily included with the plurality of shares into which the package is split. IBM explains that "the IBM COS SecureSlice AONT encryption process" involves "[g]enerat[ing] [a] random encryption key" and "[e]ncrypt[ing] data by using encryption key," "[c]alculat[ing] the hash of encrypted data," "[c]alculat[ing] the exclusive-OR (XOR) of hash and encryption key" and "[a]ppending the result to the encrypted data to create the AONT package."[83] In other words, "[t]he key is packaged along with the data"[84] Relatedly, as explained above, "[w]hen a segment of data is to be stored in a dispersed storage system, an integrity check value is first appended to the data."[85] This is shown in the exemplary IBM graphic below:[86]

[82] IBM White Paper at 17.

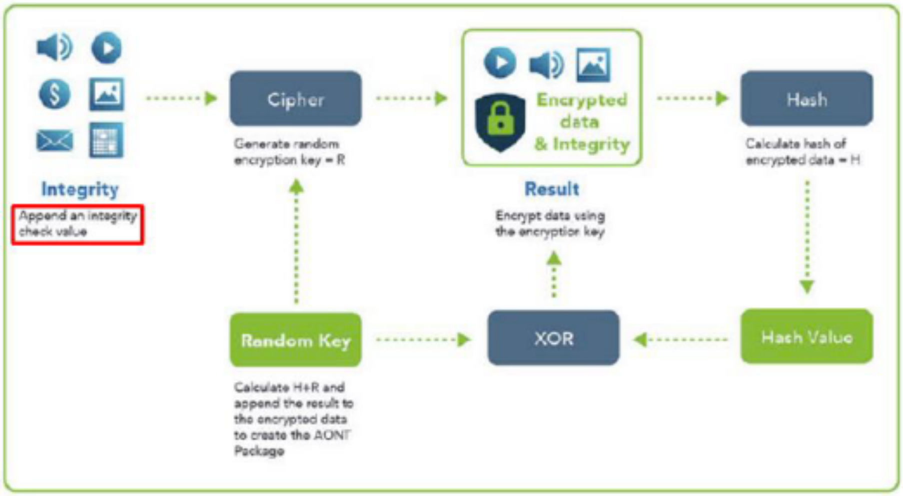
[83] IBM Red Paper at 17.

[84] IBM White Paper at 9.

[85] IBM White Paper at 10.

[86] IBM White Paper at 9-10 (red annotations added).

2. Data is encrypted using RC4-128 encryption with MD5-128 hash for data integrity. Also supported: AES-256 encryption with SHA-256 hash. The key is packaged along with the data.


Integrity
Append an integrity check value
Cipher
Generate random encryption key = R
Encrypted data & Integrity
Result
Encrypt data using the encryption key
Hash
Calculate hash of encrypted data = H
Random Key
Calculate H+R and append the result to the encrypted data to create the AONT Package
XOR
Hash Value

<u>Claim 1[f] causing the plurality of shares to be stored in respective separate storage locations;</u>

159. IBM meets this limitation. For example, IBM's Cloud Object Storage System causes the plurality of shares to be stored in respective separate storage locations.

160. As explained above, after IBM's Cloud Object Storage System splits the data into slices, those slices are "distributed to the Slicestor nodes."[87] And "[i]f the storage pool is configured to use SD Mode, each Slicestor node in a device set stores a single slice."[88] But "[i]f the storage pool is configured to use CD Mode, each Slicestor node stores multiple slices. The system ensures that the slices are not stored on the same drives within the same chassis."[89] Each drive within a chassis in a Slicestor node is a separate storage location.

[87] IBM Red Paper at 19.

[88] *Id.*

[89] *Id.* at 20.

Claim 1[g] wherein the data set is restorable by accessing less than all, but at least a threshold number of, the plurality of shares.

161. IBM meets this limitation. For example, within IBM's Cloud Object Storage System, the data set is restorable by accessing less than all, but at least a threshold number of, the plurality of shares.

162. As explained above, in IBM's Cloud Object Storage System, "[d]ata is reassembled in segments, and for each segment, thousands, if not millions, of combinations of slices are examined to determine the best delivery path."[90] And "[a]fter the read threshold number of slices [(*i.e.*, minimum)] is received, the Accesser node decodes the object and verifies its integrity."[91] "The read threshold of an IDA defines the number of slices of the width that must be available for the data to be readable. For example, if the read threshold of a 12-wide system is set to 6, the system needs only six slices to read the data."[92] In other words, the "IBM COS requires only the read threshold number of Slicestor nodes that are available to reconstruct the data."[93] Below is an exemplary IBM graphic illustrating the process[94]:

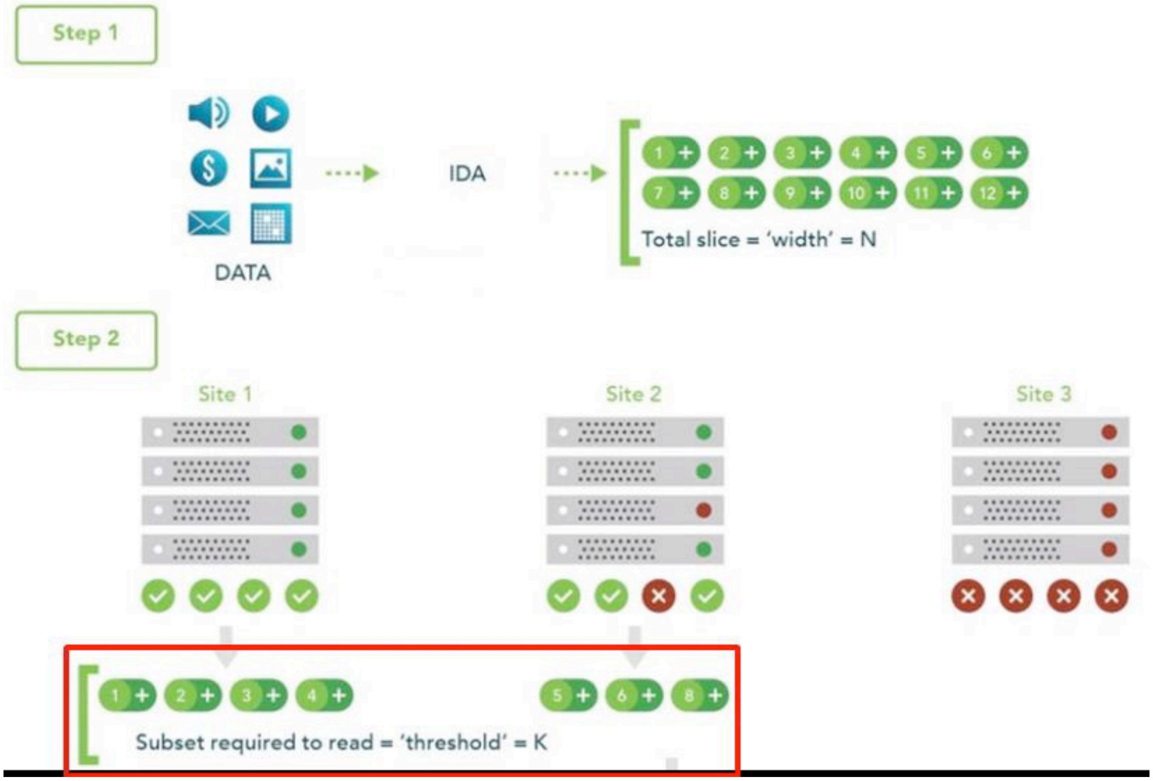
[90] IBM Definitive Guide at 14.

[91] IBM Red Paper at 22.

[92] *Id.* at 11.

[93] *Id.*at 23.

[94] IBM Definitive Guide at 5 (showing steps 1 and 2; red annotation added).



* * *

163. To the extent any limitation of claim 1 of the '802 Patent is not met literally, on information and belief, each such limitation is met through application of the doctrine of equivalents. The elements of the Accused Product are at most insubstantially different from, and perform substantially the same function in substantially the same way to achieve the same result as, each limitation of claim 1 of the '802 Patent.

164. On information and belief, IBM had knowledge of, or was willfully blind to, the '802 Patent. On information and belief, IBM had knowledge of, or was willfully blind to, the fact that its conduct constituted, induced, or contributed to infringement of the '802 Patent.

165. As a result of IBM's infringement of the '802 Patent, SFI has been damaged. SFI is entitled to recover from IBM damages sustained as a result of IBM's wrongful acts sufficient to

compensate SFI for the infringement in an amount subject to proof at trial, and in no event less than a reasonable royalty.

166. To the extent 35 U.S.C. § 287 is determined to be applicable, on information and belief, its requirements have been satisfied with respect to the ’802 Patent.

167. IBM also indirectly infringes one or more claims of the ’802 Patent in violation of 35 U.S.C. § 271(b) by actively inducing others (including, but not limited to, IBM’s customers) to infringe the ’802 Patent, by, among other things, providing, on information and belief, instructions, manuals, technical assistance, and promotional materials relating to the installation, use, operation, and maintenance of the IBM Cloud Object Storage system in the United States. Defendants’ inducement is ongoing.

168. IBM also indirectly infringes one or more claims of the ’802 Patent in violation of U.S.C. § 271(c) by contributing to others’ infringement of the ’802 Patent by, on information and belief, selling, offering to sell, and importing components that its customers use to build and assemble, and that its customers use to operate, the IBM Cloud Object Storage system, which components constitute a material part of the claimed invention of the ’802 Patent. IBM knows that the components of the IBM Cloud Object Storage are especially made or adapted for use in infringement of the ’802 Patent, and those components are not a staple article or commodity of commerce suitable for substantial non-infringing use. Defendants’ contributory infringement is ongoing.

169. SFI has suffered and continues to suffer irreparable injury as a direct and proximate result of IBM’s infringement for which there is no adequate remedy at law. Unless IBM is enjoined, SFI will continue to suffer such irreparable injury.

PRAYER FOR RELIEF

WHEREFORE, SFI prays for judgment against IBM as follows:

A. That IBM has infringed, and unless enjoined, will continue to infringe, each of the Asserted Patents.

B. That IBM's infringement of the Asserted Patents was willful.

C. That IBM pay SFI damages adequate to compensate SFI for IBM's infringement of each of the Asserted Patents, but in no event less than a reasonable royalty, together with interest and costs under 35 U.S.C. § 284.

D. That IBM be ordered to pay prejudgment and post-judgment interest on the damages assessed.

E. That IBM be ordered to pay supplemental damages to SFI including interest, with an accounting, as needed.

F. That IBM be enjoined from infringing the Asserted Patents, or if its infringement is not enjoined, that IBM to ordered to pay ongoing royalties to SFI for any post-judgment use.

G. That this is an exceptional case under 35 U.S.C. § 285, and that IBM pay SFI's attorney's fees and costs in this action; and

H. That SFI be awarded such other and further relief, including equitable relief, as this Court deems just property.

DEMAND FOR A JURY TRIAL

Pursuant to the Federal Rules of Civil Procedure, SFI hereby demands a trial by jury on all issues so triable.

Dated: March 24, 2025

By: /s/

Charles B. Molster, III
Virginia Bar No. 23613
The Law Offices of Charles B. Molster, III PLLC
2141 Wisconsin Avenue, N.W., Suite M
Washington, D.C. 20007
Cell: (703) 346-1505
cmolster@molsterlaw.com

William R. Poynter
Virginia Bar No. 48672
KALEO LEGAL
4456 Corporation Lane, Suite 135
Virginia Beach, Virginia 23462
Telephone: (757) 238-6383
wpoynter@kaleolegal.com

Andrei Iancu (*pro hac vice* forthcoming)
SULLIVAN & CROMWELL LLP
1700 New York Avenue, N.W., Suite 700
Washington, D.C. 20006
Tel.: (202) 956-7500
SFI-IBM@sullcrom.com

Dustin F. Guzior (*pro hac vice* forthcoming)
Alexander N. Gross (*pro hac vice* forthcoming)
Stephen J. Elliott (*pro hac vice* forthcoming)
SULLIVAN & CROMWELL LLP
125 Broad Street
New York, New York 10004
Tel.: (212) 558-4000
SFI-IBM@sullcrom.com

Aviv S. Halpern (*pro hac vice* forthcoming)
SULLIVAN & CROMWELL LLP
550 Hamilton Avenue
Palo Alto, California 94301
Tel.: (650) 461-5600
SFI-IBM@sullcrom.com

Attorneys for Plaintiff Security First Innovations, LLC