Data Identification Challenges and Considerations Security Measures

Anand Athavale

andyathavale@gmail.com
Independent Researcher
Decades of Industry experience in Data Management

Abstract

Data as a term gets used in so many ways and so many levels of granularity that a lot of problem descriptions, regulations, best practices and even solution descriptions make non-practioners think of it as an easily identifiable entity. However, there are numerous challenges to be handled just for referring to, or, giving additional information about data, often termed as metadata. Any attribute, information, or activity to be indicated on a data needs an identifier to accurately denote the "data" for which any of these are being captured and shown. Various data types combined with data location makes the identification issue more complex. If not handled properly, IT practitioners using different applications may not necessarily be aware, or, be confident whether the IT practitioners or the applications are in fact referring to the same "data" item. Recent developments in object storage have made this issue further invisible, but the challenges of data identification are real and those have a serious impact on the choice and effectiveness of data management applications.

Keywords: Data Identifier, hash, application integration, data transformation, data migration, data layout

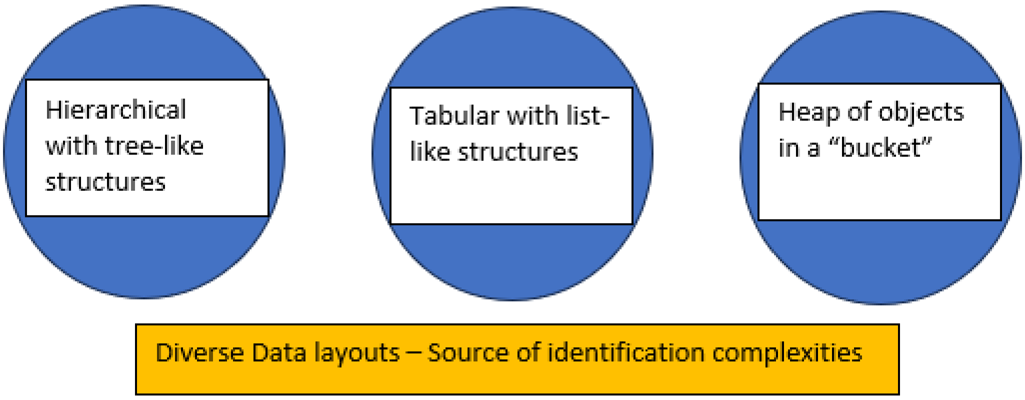
Introduction

Data has been around for years. The digital data is more recent but the paper-based data was there much before. The types of the paper-based data were subdivided into different purposes and formats. A book, a thesis, a project report, ship records, entry-exit registers were the different purposes and text-based books, charts and maps were some of the formats. But everything had a clear identification label. "High school senior year project report for X topic by John Doe", "To kill a mocking bird" novel, "The Titanic passenger list" with embarking port, where it was headed and the date of departure, all had identifiers to distinctively represent those and when referred that way could be uniquely identified. The digitization also had the same concept for some of the data formats and data types, but these identifiers went inside of the content and were not mandated to match the storage level and layout driven identifiers they had. For example, a file name doc1.doc can contain anything ranging from a financial report to a nicely written article or a book chapter, or a whole book. Of course, the possibility of same happening with a bunch of pages cover page saying "Nanna's appointment" while the inside pages contained grandmother's secret recipe. But such occurrences were rare.

Current data technologies, especially the one for unstructured and semi-structured data, have not paid much attention to the ease of reference for data management applications. They do maintain their internal unique identifiers, but those primarily serve the data storage application along with any peripheral applications built by the same vendor for managing the data beyond just storing it. But when it comes to wholistic consideration of the various stages, transformations and migration data can go through, the identification does not seem to be conducive and open enough to keep the lineage [1]. Of course, a lot of technologies do have additional attributes for users of data to record what they want to add as a label, description, or similar notations. But the issue is that those attributes are not treated as unique identifiers, especially when referring to it from outside. Moreover, the identifiers are mathematical and codified making it difficult to the IT or security practitioners to know what that data item really is just by "looking at it." These are the complexities of data identifications in the modern digital world and while appearing to complicate the usage, they can have significant impact on security measures and the choice of data management applications.

Data storage and layout impact on data identification

Common types of granular data types are files which get stored in various storage types like Network attached storage on what typically is referred as shares on storage arrays or file servers, or, on semi-structured technologies like SharePoint underneath using a database like SQL Server, or, Software-as-a-Service applications like One-Drive, SharePoint Online, Box and Google workspace. These variations have some unique behaviors which create challenges for data identification for IT practitioners and application owners.


Hierarchical with tree-like structures
Tabular with list-like structures
Heap of objects in a "bucket"
Diverse Data layouts – Source of identification complexities

i. Hierarchical Behaviors of NAS systems

A share on a network device has relatively straightforward concept called UNC path [2]. It is designated as \\fileserver-name\share-name\[directory hierarchy to the level containing file]\[filename]. For any data user, IT practitioner and a relatively established data management application, this works for identifying a file uniquely. But thereare still some considerations to it. As an instance, the underlying file server could be a Windows file server with name WINFS1with say "E" Drive as the base of the share created on a folder "E:\shared-data\folder1" but mapped as \\WINFS1\share1. So while on the file server, a file E:\shared-data\folder1\child1\file1.txt would appear as \\WINFS1\share1\child1\file1.txt. For any data access control reporting tool, it is vital for giving this reference because a data user may access and report any change request as \\WINFS1\share1\child1\file1.txt but the storage administrator needs to be aware that the file actually is residing on the file server WINFS1 E drive under folder E:\shared-data\folder1\child1. One may think that because the storage administrators would remember it because they created it but imagine scenarios with hundreds of file servers with thousands of shares, hundreds of thousands of folders and millions of files.

Now, the same file1.txt under a folder child1 does not follow the same path convention on SharePoint. A SharePoint URL, for the lack of better word, will look like https://<webapp>/sites/<sitename>/[SharePoint specific internal path denotations]/Doc.aspx/[SharePoint translated ID and encoded file name]. A box file URL could look like https://*.app.box.com/files/<unique file identifier>. This is enough inconsistency to cause complexity and confusion in just ensuring that the data item is being referenced correctly when giving any information about that data item, like access permissions or sensitivity labels.

ii. Single path as multiple shares, DFS and links

NAS technologies allow the same path to be used for multiple shares. The example earlier where E:\shared-data\folder1 is shared as \\WINFS1\share1 could also be shared as \\WINFS1\sh1. To make matters further complicated the underneath folder child1 with original file server path E:\shared-data\folder1\child1 could also be shared as \\WINFS1\chshare1. Now, the file1.txt in earlier example is accessible with three different paths while from the storage perspective, it is the exact same file. However, due to share level permissions, the file1.txt can have different access permissions for different UNC paths. This situation is rare but not impossible. Microsoft also has a concept of Distributed File System, or DFS, which is a Microsoft implementation of a network file system that allows data to exist in multiple places over a network while mitigating the need for clients asking for such data to know exactly which server and path the data exists upon. Now while this simplifies the access for end users where they see a path as \\Domain\commonfiles\[Inside path for file1.txt], the applications reporting on access permissions or sensitivity labels need to be aware of these, to make the life of storage administrators, or, security auditors easier by not having to remember which DFS path means which physical location of the file.

iii. Cloud/On-premises Object Storage hierarchies and services

Traditional file systems follow a hierarchical model for arranging data into shares or similar concepts, then folder trees and files under folders. Object storage disrupts that norm and literally follows concepts of a “bucket” with several “objects” in the bucket. While there are ways to visualize the data arranged as hierarchical layout, many a times it is simply objects put into buckets. Here the problem is somewhat opposite. An object has too many attributes which could be used for reference or use. A key could be deemed as a name. But, for real uniqueness, a key is paired with a version ID. Besides these, the URI could have different styles. Additionally, there is a concept of tags. A tag is a keypair consisting of a key and an optional value which you can use to categorize how the bucket or object fits into an organization.These variations cause complexity for end users and the IT practitioners to have some sort of understanding on how to denote or reference an object when exchanging, reviewing, and monitoring data within the object storage. While databases are treated differently, when the security attention will turn to databases, the diversity in terminology and layouts in database technologies will also cause similar challenges.

Data type impact on data discovery and identification

Data discovery at a very generic level is starting a specific point of data storage to then build out an inventory closest to how the data is arranged. Data discovery for a traditional unstructured data source like a NAS or OneDrive is still relatively straight forward. The discovery application could start at the NAS server level, discover shares, and then discover the hierarchy within the shares with folders and files and almost build an windows explorer like hierarchical view. For One Drive, the application could start enumerating each account under an organization, not considering the access or permission issues. But when it comes to discovering and associating data within more modern concepts like teams, slacks and similar applications, the lack of supportability within the applications and the absence of much of a hierarchy creates an issue for discovering data and then representing it in a meaningful and consistent way for the storage, security, and backup admins to collaborate. This also creates challenges with uniquely identifying any data item. If the same data item has multiple copies with added numerical value to create another instance to be treated as the same item, or, is to be considered separately. If the data lineage is to be tracked, then where would be the origin of the data item. Would it start within the application specifically in question like MS teams, or, should it be considered a SharePoint or One Drive account where it may have been originated. Moreover, how to then give a single unique identifier which could be understood by various IT teams and applications for referring to it?

These issues do not present themselves for every data item. They are more pronounced for those data itemsinvolved in data loss, data leak or data tampering situations. While that, IT teams cannot start pondering on these reference schemes when the need arises. They must be ready with the naming understanding much before that.

Considerations for choosing data discovery and monitoring solutions

i. Application awareness

Data discovery and monitoring solutions need to closely understand the nuances of the data source applications to reduce the room for interpretations and mistakes. Each application refers to data items differently including the hierarchy. Data monitoring solutions which take a short-cut and base the naming on either one or the other application naming like file servers and shares, or, invent their own naming to avoid dealing with the application specific naming conventions should be avoided.

ii. Role and interaction awareness

It is crucial for the data discovery and monitoring solutions to understand the difference in interaction and in turn the referencing conventions for the service providers and the consumers of the service, for any data source application. As an example, the end user of data may refer to a path as Distributed File System convention with root being the domain followed by the "easy to remember" hierarchy [3]. But the administrator needs to be shown the end user view and the physical storage view to easily identify data items in the question, for which either a sensitivity label or any access permission issue is being flagged. Some of the data protection applications have started providing data discovery and monitoring capabilities. However, advanced data protection application sometimes changes the data storage completely for efficiency and resiliency purposes, which eliminate any direct way to reference a single data item [4]. Most of the times, those provide a way to visualize those as if stored on a layout same as the data source applications. But, unless the translation is provided for the data protection administrators to map the data item to a the granular most storage unit as per the data protection storage systems and applications, they cannot implement security controls in an effective manner. Moreover, often these data protection systems become a black box for security teams trying to ascertain the effectiveness of security measures inside the data protection systems and the solutions. This problem is same as denoting accurate address of a person or an entity. C/5432, Rosewood complex, Flower Lane, Mooncity, PB, 765442, NewGenCountry sufficiently pinpoints to the location as C probably being the building or a wing, 5432 being a unit number, Rosewood being an apartment or a building complex, Flower Lane being the road, Mooncity being the city, PB denoting a well-know state, or a region and NewGenCountry referring to the country itself. If, however, the address for a personwas only available to city name, like,"Mooncity" in NewGenCountry, it will be impossible to pinpoint the location for a person or an entity.

iii. Visualization awareness for practical challenges

Data Discovery and monitoring solutions need to be accurate and it is true for any applications. However, that is just a necessary condition and not bring sufficiency to the effectiveness. Often, the data reference mechanism like UNC path could be very long. Human brains are not equipped to process long paths. Hence, these solutions need to go passed displaying the paths as is, and then the user interface technologies and real estate limit the usability of these solutions. Human brains are also not equipped to process URLs, especially which may not always contain meaningful text to clearly map and understand to being useful. The data source application provides other naming attributes, but those attributes may not always be considered unique identifiers. This is the complexity which needs to be handled by data discovery and monitoring solutions. Many applications resort to simply create unique hashes to identify a data item as duplicate or unique. But, barring some exceptions, human brains are not equipped to memorize long numbers or hexadecimal characters [5]. Just think of this example. Imagine in a large country such as the United States of America, all the license plates were to be unique across the country. While current license plates are around 7 alpha-numeric long and those itself create challenges to remember, if those were to be 12 alpha-numeric long, how much more difficult it would have been? This is why the data discovery and monitoring solutions must consider these human aspects. These solutions may take a route of being autonomous and carry out most of the tasks in an automated way. However, the respective storage, application and security professionals still need to be able to review and cross check. So simply automating everything will not be sufficient.

Conclusion

Data as a term gets used loosely for describing a data related problem and a solution. However, the specificity in referring to a data item has become a challenge due to proliferation of data source technologies and applications and complexity of layout and hierarchies, or in some cases, absence of those. This challenge is often either goes unnoticed, or is brushed aside to avoid slowing down building of any solutions. The balance of innovation in making a solution smart vs. considering the complexities of application diversity and limitations of human brain is overly skewed towards the smartness. This lack of focus on data identification creates security risks besides usability challenges. Storage, Application and Security practitioners need form an alliance in demanding that such solutions address this challenge not for one, or the other role but for all the parties involved.

References

[1] Itamar Ben Hemo, Former Forbs Council Member, Forbes Technology Council Council Post, (March2021), https://www.forbes.com/councils/forbestechcouncil/2021/03/25/data-lineage-you-cant-trust-what-you-cant-see/, (August, 2021)

[2] Jordan Borean,“Windows mapped drives – what the h*** is going on?” (November 2018). https://www.bloggingforlogging.com/2018/11/22/windows-mapped-drives-what-the-hell-is-going-on/, (September 2021)

[3] D. Patil, My experience with Windows Distributed File System and Replication(DFS-R), (December2018), https://dpatil1410.wordpress.com/2018/12/17/my-experience-with-windows-distributed-file-system-and-replicationdfs-r/, (September 2021)

[4] Nick Cavalancia, The role of Granular Recovery Technology in backup strategy (March2016), https://www.n-able.com/it/blog/the-role-of-granular-recovery-technology-in-backup-strategy, (September, 2021)

[5] Leon Ho, Human Brains Aren’t Designed to Remember Things, (July2017), https://medium.com/the-mission/human-brains-arent-designed-to-remember-things-1074365f0da2, (September 2021)