DATATRUST
SOLUTIONS

Secure Archive Manager

Introduction

Data growth and adhering to compliance requirements are the fundamental challenges facing organizations today. With increasing regulatory and compliance requirements, managing data is expanding beyond the domain of IT and storage administrators and risen to c-level management. Organizations must establish and administer Policies for data retention, legal hold and data disposition. These Policies are established and must be set and managed over the defined life cycle of a data set, which typically exceeds the life span of the hardware hosting it – whether in a data center or the cloud.

End-user productivity is a major business driver of data growth. Employees are accustomed to managing their day-to-day work with email, documents, images and reports. The users of data and IT organizations are engaged in a constant tug of war. End users want unlimited storage and fast access to their data from any device anywhere. IT managers, with limited budgets, want to reduce the burden on storage infrastructures and storage management. This problem is magnified by the fact that over 80% of data is inactive but continues to reside on its original T2 or T3 storage location and consume expensive storage capacity.

Forward Design

Cloud and object-based storage is continuing to be the revolutionary path for IT organizations to modernize their storage and workflows for new applications. The reasons for the transition to a new approach include: ease of access, lower cost, more data protection options, service availability, and scalability. Storage should enable organizations to grow and scale with their business needs seamlessly and never be a barrier to innovation and productivity. Storage technology is and will continue to evolve with each new generation designed to add features, improve availability and scalability, while reducing cost. DataTrust (DTS) understands these emerging technologies will enable businesses to move forward and we designed in support for them in our new data management software appliance called Secure Archive Manager (SAM).

At the same time many applications are firmly entrenched in environments using legacy CIFS and NFS file systems. These organizations would like to take advantage of cloud and object storage but the applications are not ready to enable the leap. SAM meets the current needs of the applications by providing traditional file system interfaces. SAM also accepts file system interfaces and places the data into cloud and object storage systems, while preserving all the original meta-data. Thereby enabling organizations to start the journey to cloud and object technologies with storage and let the applications catch up. SAM has optimizations designed specifically for cloud and object storage. SAM also provides the Amazon S3 front-end access protocol, to help the customer to transition to end-to-end cloud or object when the applications are ready.

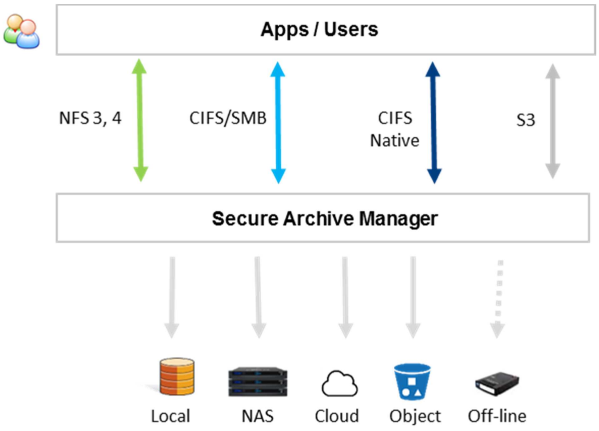
For legacy environments not transitioning to cloud or object SAM delivers technology advances with virtual file systems. By using Filter & Fuse drivers SAM decouples the file system ball-and-chain from the Operating System and its limitations. Virtual file systems can scale to billions of files, span storage systems, and span geographic locations. All while helping organizations to meet their compliance requirements.

SAM can run on a physical server, in a Virtual Machine or in an EC2 instance. As organizations transition from the physical to the virtual environment in the datacenter, SAM can easily adapt. As organizations transition the datacenter to the cloud, SAM can make the journey as well.

Secure Archive Manager

SAM is a software appliance that enables flexible data management over the life cycle of the data, independent of the access protocol and storage technology. SAM allows legacy applications to read and write to CIFS (SMB and native CIFS) and NFS (v3.x and 4.x) shares. SAM allows more modern applications to write to an S3 interface, whether that is today or in the future.

Figure 1: Secure Archive Manager Access

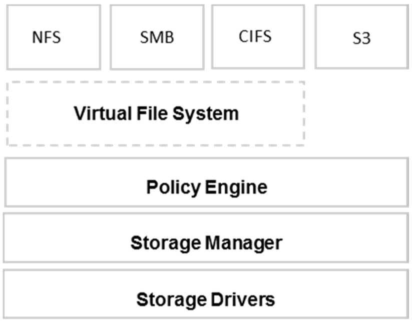


Power of Virtualized Storage

SAM leverages Virtual File Systems to abstract the view and access to the file system from the actual location of the physical content. This means that the user or application connections to data never need to change. Data can be moved from one storage system (e.g. NFS/NAS) to a new shiny converged infrastructure appliance or alternatively to a cloud or object system without any changes required to the user or application's file system. Data is no longer tied to storage hardware or its operating system, it is fluid. Now data can be placed in a storage location where it can benefit the organization the best, whenever the need arises. The placement of data is handled by Policies and performed by an internal storage manager.

SAM uses Virtual File Systems (VFS) in place of Operating System physical file systems. Operating System file systems have limitations in file counts per folder or directory, challenges with handling odd characters (e.g. emoji or foreign character sets), file system scalability (e.g. inode limitations), global name space, etc. Virtual File Systems use special drivers that intercept the OS file system calls and route them to SAM for processing. The "file system" in SAM is a MYSQL database, so it is not subject to restrictions like OS based file systems. For NFS or SMB shares a FUSE driver is used on SAM running on Linux. For native CIFS shares a special Filter Driver is used with Windows Server 2008/2012.

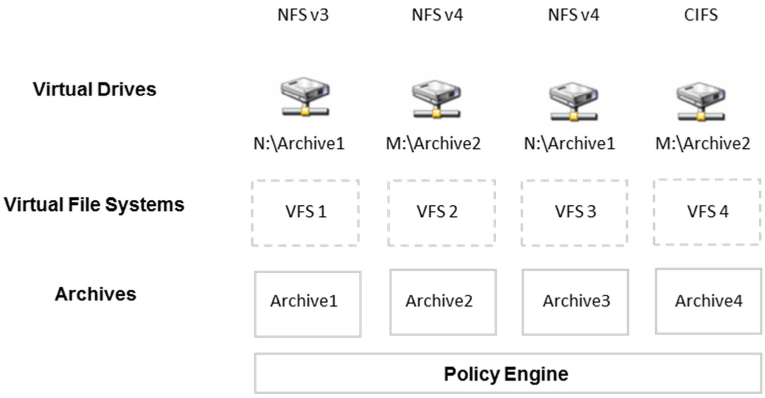
Figure 2: Secure Archive Manager Environment



Front-end Access via Shares and Buckets

The atomic unit in a SAM environment is an Archive. Each Archive is autonomous and an independently managed entity thus providing a multi-tenant ecosystem. When the SAM administrator creates an Archive a dedicated Virtual File System is created for the Archive. The administrator can configure the Archive for file system protocol access using: NFSv3, NFSv4, SMB/CIFS, or native CIFS. Alternatively access to the Archive can be configured for the S3 protocol. In this case the Archive is referred to as a Bucket to be consistent with cloud nomenclature.

Figure 3: VFS and Archives



The administrator can expose the Archive Shares for normal usage or can take them off-line. Each Archive is an independent Virtual File System that is governed by administrative configuration settings and Policies. SAM provides a multi-tenant environment for VFS / Archives that can have distinct Policies and Storage or the Policies and Storage can be shared.

Archive Configuration Options
Each Archive can be independently configured to meet the needs of the organization. The following options are available:

Access Protocol – NFSv3, NFSv4, SMB/CIFS, Native CIFS, S3

Archive Type – Production or Test

Operating Mode – Read/Write, Read-Only or WORM

Share Visible – Yes, No

Compression – Yes, No (uses LZ4 compression)

Hash – options include MD5, SHA-1, and SHA-256

Containers – option to aggregate small files based upon file count, container size or time

Encryption – each file is encrypted with a unique AES-256 key

Deduplication – is done at a file level, this is sometimes also referred to a single instancing, within an Archive

Replication – can only be enabled when another SAM system is used

Retention – there are many options and they will be covered in the section on Policy

SAM administrators or application owners may want to experiment with Archives and the various configuration options. Rather than setup a sandbox environment, SAM provides a new type of Archive called “Test.” The key advantage to Test Archives is that all the data can be deleted, regardless of any Retention or WORM settings. Test Archives cannot be converted into Production Archives.

Containers are an additional feature that allows SAM to collect a group of files and save them to storage as a single file. Container options include file count, capacity or time. Containers are especially important when an Enterprise Content Management (ECM) needs to archive large number of files to NAS or cloud storage. ECMs are famous for consuming all the inodes of a NAS when only 30% of the capacity has been used. Clouds charge for API calls so packaging up 10,000 small files into a container is more efficient and cost effective than managing them individually. Some clouds, like Amazon S3 IA (infrequent access) have minimum object size of 128KB so sending smaller objects waste capacity (e.g. 10KB file size consume 12X recorded capacity) and drive up costs (e.g. $0.0125 per GB/month becomes $0.15 per GB/month, which is multiples of than S3 standard $0.025).

Policy Driven Actions

Policies can be applied at ingest (e.g. set retention), when data is at rest (e.g. legal hold) or to manage storage (e.g. Containers or Storage Groups). Policies are defined independent of an Archive and can be applied to more than one Archive.

Policies applied to data at rest include:

> ***Legal Hold*** – A Policy applied to data at rest that does not allow a user or administrator to delete or modify data for a specified time period. If Legal Hold Policy is applied there is an option to automatically delete expired files.
>
> ***Discovery Hold*** – A Policy applied to data at rest that does not allow a user or administrator to delete or modify data for a specified time period while the data is under review for Legal Hold. Until the eDiscovery features are released in SAM version 2.0 this feature will be the same as Legal Hold.
>
> ***Read-Only*** – A Policy that is applied at an Archive level (e.g. all files in the Archive) that restricts user and administrator privileges to only enable reading of files. No files can be ingested, modified or deleted.

Policies applied to data at ingest:

> ***Retention*** – A Policy applied at an Archive level that does not allow a user or administrator to delete or modify data until after a specified time has elapsed. The Retention Units can be defined in seconds, days, months or years. The Retention Policy has two sub-policies: *Grace Period* and *Data Deletion on Expiration.* Associated with the Retention Period is a Grace Period. The Grace Period allows a predefined window of time after file is ingested prior to the setting of the Retention Policy. During the Grace Period a file can be modified or deleted by a user or administrator. The Data Deletion Policy enables file to be automatically deleted after the expiration of the Retention Period or SnapLock period. The default setting is to NOT DELETE files.
>
> ***SnapLock©*** – A Policy applied at an Archive level that uses a per file meta-data value to set the Retention Period of that file. If the "access time" meta data value is set to a future time then SAM will place the file under Retention and not allow the file to be deleted or modified until after the access time date has been passed.

Containers – A Policy applied at an Archive level that stages files until a parameter (e.g. file count, file size or time) is reached, then the collection of files are placed into a single container file which is then written to storage.

Storage Policies include:

Storage Copies – define a storage group consisting of two or more storage locations and SAM will write a copy to each location

Storage Pooling – define a storage group consisting of two or more storage locations and SAM can fill one and the start writing to the other or SAM can write in round robin fashion to the group.

Cloud Writes – the default behavior is that SAM writes to the Cloud storage when files are ingested, however options to schedule to off hours and to throttle bandwidth are available.

Cloud Cache – for hybrid environments this option provides for configuration of a local cache and local retention and cloud copy options.

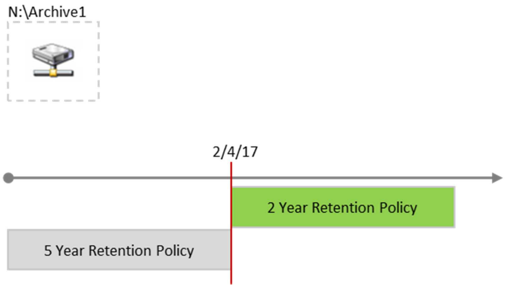
Tiering – Not supported until version 2.0.

Off-Line - Not supported until version 3.0.

SAM allows Policies implemented on an Archive to be suspended and new Policies to be implemented.

Example: Archive called Email assigned a Retention Policy of 5 Years. Three years later on February 4, 2017 the Chief Compliance Officer changes the Email Retention Policy to 2 Years. The SAM administrator of the Email Archive would suspend the 5 Year Policy. They would then define and implement a 2 Year Retention Policy starting on February 4, 2017. This means that data ingested prior to February 4, 2017 will be under Retention for 5 Years from the original ingest date. New file from February 4, 2017 will only be subject to a 2 year retention date.

Figure 4: Adjustable Policies



Administration

SAM can join a Domain and be managed by Active Directory or LDAP. Alternatively SAM can be managed by a Global Administrator who can delegate Administration of an Archive to an Archive Administrator. SAM also supports global and Archive specific users. Administrator or Users can elect to receive email alerts on SAM activities based upon level of criticality. There is a third user class called Auditor, which has read-only access to content. This role is useful during eDiscovery processes.

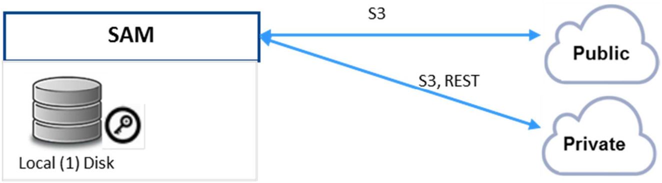
Deployment

Secure Archive Manager is extremely versatile and supports a wide variety of deployment options. SAM was designed to run in a physical or virtual environment. For organizations requiring NFSv3, NFSv4 or SMB/CIFS protocol support SAM will run in a Debian Linux environment. For organizations requiring native CIFS and/or Azure Support SAM will provide these services on Windows Server 2008 or 2012.

For hybrid Cloud environments writing to Microsoft Azure SAM needs to run in a native CIFS / Windows Server environment. SAM can stream directly to Azure or SAM can stage data into local disk cache and then write to Azure on a schedule.

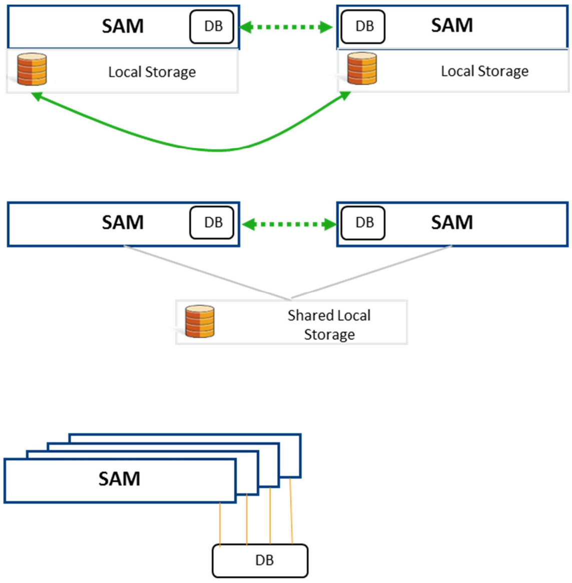
For hybrid Cloud environments writing to a Cloud other than Azure SAM can run in a Linux or Windows environment. The Cloud Write Policy is used to control when data is written to the cloud or to throttle bandwidth when writing to the Cloud. The Cloud Cache Policy can be used to retain a local copy for a period of time.

Figure 5: SAM and Cloud



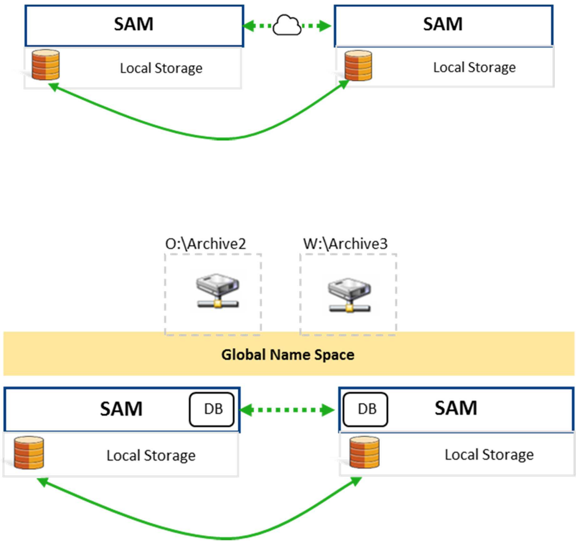
For organizations requiring higher levels of availability SAM can be deployed in an Active-Active or Active-Passive pair. SAM can also be deployed in a 3 node or larger cluster will all nodes being Active.

Figure 6: High Availability Options



For organizations requiring DR or geo-dispersion SAM can replicate to another instance of SAM. The replication is asynchronous, meaning that the remote SAM does not have to commit before the local SAM reports a file has been archived. If ZFS or Ceph is used as the storage option for both locations the storage replication will be handled at the storage layer.

Figure 6: SAM and DR



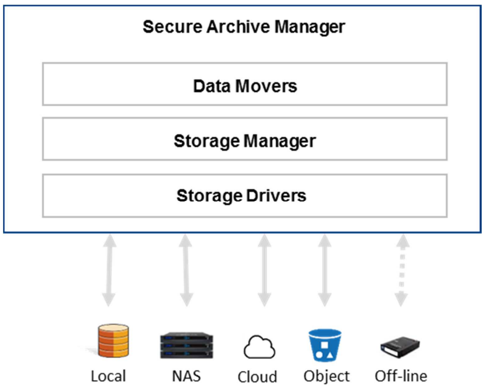
System Requirements

SAM was designed to run in a physical environment or virtual one (VM, KVM, or Hypervisor). The minimum specification for a host in a small (under 100 million files/objects) NFS environment is: 4 core CPU, 8GB RAM, 100GB disk, 2x1 Gb Ethernet (includes one port for management). For a small CIFS environment (under 100 million files/objects): 8 core CPU, 16GB RAM, 100GB disk, 2x1Gb Ethernet. For better performance upgrade the RAM first and then number of CPUs.

Storage

SAM supports local storage, networked storage, cloud and object storage. Local storage is anything that the host running SAM can see as local storage. This includes local disk, iSCSI, SAN, and network drives.

Figure 7: SAM Storage



SAM also supports network storage such as NAS, Cloud and Object. Another key concept is SAM is Storage Groups and associated policies. A Storage Group to consist of two or more storage locations (e.g. local storage and cloud storage, or NAS and object storage). A preconfigured Storage Group Policy will direct SAM to save a copy of ingested content to all locations in the Storage Group. An example would be to save content to local disk and immediately make a copy to Amazon S3. Or the S3 copy could be scheduled to off time such as 1AM. In addition to making copies, a Storage Group has a read hierarchy. Continuing with the above example if a file was not available on local disk SAM would not return a read error to the user or application. Rather SAM would automatically try and retrieve the file from the second (e.g. Amazon S3) or third location (e.g. DR site). By transparently serving failed requests SAM avoids calls to IT resources to fix the problem. In the cases of writes, if the primary location is not available then SAM will write the data to the next storage location. In all cases an error is logged for review. For writes to secondary locations SAM has an integrated process that logs a copy was not written to the primary storage location and then once the original storage location is back on-line SAM will make a copy from the alternative location and log this action.

For organizations requiring higher levels of data protection than traditional RAID or mirroring, SAM supports ZFS and Ceph. Both ZFS and Ceph have robust features for identifying bit rot and self-healing capabilities. They can also replicate data from one SAM instance to another and keep the data in synch. We recommend ZFS for smaller installations, which include two parity disks, so the minimum configuration is six drives. Ceph is best when run on separated nodes and due to the vast number of configuration options it is best to schedule time with a technical resource to understand your requirements so an appropriate configuration can be provided.

Policies for multiple copies, pooling, tiering, off-line and cloud copies all rely on an internal data mover. Policies schedule the copy of data from one storage location to another. After a copy has been made it is verified by comparing content based cryptographic hashes on the source and target. Copies to the Cloud are slightly different and vary by Cloud vendor, but the concept is similar. For copies to Amazon S3 SAM embeds an MD5 hash into the header. S3 sees the MD5 hash and calculates an MD5 hash on the ingested object contents. If the MD5 hashes match the S3 saves the file, if not it rejects it and SAM has to retry the copy. Glacier is more involved and requires a discussion with a DTS technical resource.

External Data Movers

In SAM version 2.0 the data movers will provide additional functionality by moving data from external sources into SAM. This includes a one-time move where SAM can copy data from a NAS, Cloud or Object storage to a SAM Archive. This one way copy is verified using content based cryptographic hashes. This functionality is coming from our existing data migration software. The second external option is what we call T-Linking, think of this as a remote tiering option. Policies are defined for scanning the source file systems on a periodic basis with restrictions, similar to how backups run. If data is found that meets predefined criteria (e.g. file age, last access, user, group, file type or file size or a combination of factors). SAM will copy the file to a SAM Archive, then verify the copy and replace the file with a stub or link. SAM also imports all permissions and metadata. Most file systems will refuse to honor permissions on symbolic links so SAM assumes this responsibility.

Data Security

Ransom Ware and Viruses have made data security top of mind in all storage decisions independent of the storage type or storage location. SAM has excellent features to protect the data it manages locally, on the network or in the cloud. The first aspect of data security is protecting access to data.

> SAM is more than a data management application it is a software appliance or appliances in the case of cluster deployments. The first line of defense is an integrated firewall with many traditional firewall features. The firewall verifies that the traffic is of the expected type. If a Doman user attempts to encrypt files, they get automatically blackballed and locked out until after administrative review.

The second line of defense is that SAM uses Virtual File Systems, not physical file systems. So attacks targeted at file systems are ineffective since there are no OS file systems to attack. SAM intercepts file system tasks from the OS file system stack. SAM routes the requests to a database that represents the VFS.

Encryption is the third line of defense. Via Policy data archived to SAM can be encrypted at rest and in transit to the cloud or other network storage. SAM utilizes AES-256 bit encryption to encrypt every file in an Archive with encryption turned on. Note keys are pre-generated as to not impact performance.

Obfuscation is the fourth line of defense. Files can be converted into objects referenced by GUIDs, thus removing any reference to the source file name path. The Container feature combine lots of files into a single proprietary file or object that must be opened and navigated before a file can be recalled. Also encrypted files can be put into Containers. Each of these options removes any reference to the original file's name and path, thus making it impossible to tie any front-end file/path structure to anything in back-end storage.

The SAM database is critical to provide users and applications access to data protected and managed by SAM. The brains of SAM are in a database and configuration files and it is imperative that these items be backed up. SAM provides integrated backup functionality or third party applications can be utilized.

The second aspect of data security is protecting the content itself. With SAM both the access/permissions to data and the data can be independently protected with a variety of strategies depending upon the deployment model and storage options. SAM supports local clusters and a global name space across geographic locations, in addition to backup options. Thereby protecting the metadata needed to access content archived to SAM and the end storage locations and pathways needed to retrieve the content. Protecting the data can be done by mirroring drives in a VM, using ZFS or Ceph. The data protections options are many and a discussion with SAM engineers would help match your organizations unique needs to the appropriate SAM options.

The third aspect of data security is verifying the integrity of the content. SAM uses content based cryptographic hashes (e.g. MD5, SHA-1, SHA-256) to verify the integrity of every file. SAM offers both read and write verification options. SAM also conducts a periodic audit of the stored content to check for bit rot. If a file is found to not match the original hash value then a copy from another storage location is retrieved and saved, after which the VFS is updated.

The fourth aspect of data security is making the data tamper proof. This is accomplished by two processes. First, SAM creates audit logs for all User and Administrator access and actions. The integrity of the Audit logs are maintained by using content based hashes. The contents of the logs can be protected by encrypting them with a separate certificate than used for data. The logs can also be written to a location other than SAM. The second process is locking down the capability of the Root User. SAM does not allow the Root User to delete files or modify Audit logs.

Compliance

SAM was designed from the ground up to help organizations in regulated industries meet compliance and governance requirements. Many regulations like SEC17a-4 require immutable storage during a retention period, multiple copies, verified write process, serialization, indexes, audit logs, restricted root capabilities, etc. Previous sections of this white paper have covered how SAM meets these requirements, but for the full monte on how SAM meets the requirements just ask for a regulation specific whitepaper.

Cloud

SAM supports the Amazon S3 protocol, some variations and REST to write to Cloud storage. SAM provides enables content to be streamed directly to a cloud for NFSv4, native CIFS and S3 front-end protocols. NFSv3 does not have the notion of a file close so it must be staged prior to writing to a cloud. SAM uses a configurable write-cache for this purpose. Due to bandwidth restrictions or other reasons data may be staged in a write-cache and then scheduled to be written to a cloud. SAM also provides the ability to manage the network bandwidth consumed by SAM.

Hybrid deployments of SAM allow data to be saved locally and written to a cloud or clouds, immediately or on a schedule. Policies are used to configure the amount of local storage, life-cycle on the local storage and recall operations.

Cloud volumes can get hot if too much data is written to sequential space (e.g. App, Year, Month, Day, Doc ID 1, +1). SAM provides the option to disperse the data to lessen the load on any cloud resource and improve write and read performance.

SAM also writes content based hashes into the S3 header, which can be used by the Cloud to verify the data. For example if a MD5 is put into the S3 header and the object is put into Amazon S3 cloud storage it will be rejected by S3 if the cloud computed MD5 hash did not match the supplied one.

SAM also writes Windows ACLs and other permissions to the meta-data in the S3 header. This allows the original ACLS to be restored in the case of a source failure. Note Windows ACLS can extend beyond the 4K boundary and SAM accommodates this by creating additional linked objects.

Organizations can use SAM to take advantage of cloud storage today. When the organizations applications are ready to use S3, SAM provides a front-end interface SAM. To make this process work an export form SAM provides a mapping table of the original file names and path structures to the S3 object ids. This can be imported by the application owner or DataTrust Solutions can provide a tool to update the application database via direct injection. Alternatively, SAM can provide an export and once the application owners are satisfied SAM can be removed from the data path.

For cloud storage with API calls for advance retention or WORM features, SAM can interpret data and pass the values on to the end storage.

Object Storage

For the sake of brevity how SAM works with object storage is very similar to how SAM works with cloud storage. For more details arrange a discussion with DTS SAM experts.

Summary

SAM is a “data management appliance” with an ecosystem that provides better security, data protection and scalability over traditional data management software. SAM is designed to help your organization manage data over its lifecycle, independent of any storage protocol or hardware. We are storage agnostic and leave the task of managing the hardware to the makers of the hardware. SAM preserves the User and Application view of their data over the lifecycle of the data which include many transitions to new back-end storage locations. SAM provides integrated data movers that make the process of transitioning data from any storage location to any other storage location, independent of the original and target protocols or technologies. SAM provides features and administrative controls to help organizations meet governance and compliance requirements, including chain-of-custody reports as data transitions from one storage location to another.