Intelligent Archiving Whitepaper

Table of Contents

Introduction

Scout Archive Manager (ScoutAM) is a next generation mass storage platform for cost effectively managing large data collections on any mix of storage devices and cloud services.

ScoutAM functionality extends beyond file or object gateways by providing a powerful and intelligent policy engine that can be customized to serve the unique storage needs and workloads of large organizations. Through a standard POSIX interface ScoutAM accepts unorganized, random traffic from users or existing backup solutions and enterprise applications, then the policy engine uses rich and flexible data management policies to automate and optimize the flow of data both in and out of cloud storage services and mass storage devices including tape libraries, making it possible to achieve high streaming data rates of approximately 10GB/s per ScoutAM node, depending on the workload.

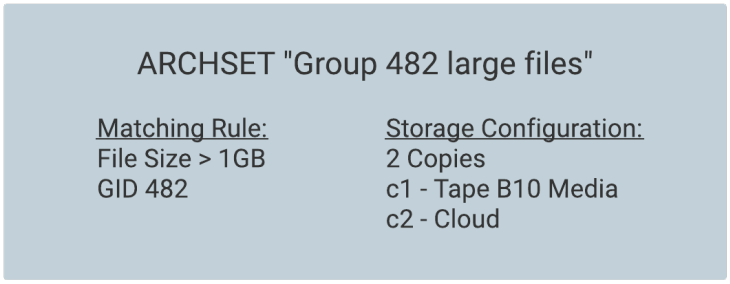
The ScoutAM policy engine not only enables the efficient grouping of data to optimize overall system utilization and throughput, but also simplifies the management of a large archival storage system with a rich set of policies and storage configurations that allow system administrators to automate and optimize storage strategies by stakeholder, by data type, by age, and by file size. Not all data classes require the same level of protection or the same recall speed. Policies can easily differentiate classes of data and automatically determine how many copies should be created on what type of media or cloud service. This flexibility enables organizations to use a single ScoutAM system across many departments and many different data sets with diverse data protection requirements while intelligently utilizing different types of storage media to optimize for performance, reliability, and cost.

ScoutAM policies allow organizations to use a single solution to manage any combination of cloud providers and local mass storage devices such as tape libraries, low-cost disk arrays, on premises object storage systems, and public cloud storage services. As data protection requirements and workloads change over time policies can be added, removed, and refined to evolve with the needs of the organization. The policy engine in ScoutAM makes it possible to maintain independence from storage hardware vendors and cloud providers, because new policies can easily be deployed to automatically utilize new back-end storage systems, or to transfer copies from one system to another.

How the ScoutAM policy engine works

At the core of ScoutAM's orchestration capability is the policy engine which enables system administrators to define custom storage rules that fit the unique needs of their organization. When a file is ingested by ScoutAM it is evaluated by the policy engine to determine how and where it should be stored. Policies can be defined to match a file based on the user and/or group the file belongs to, the name of the file, the size of the file, and the time the file was created or last accessed. When a file matches a policy definition, the storage configuration associated with that policy is automatically applied. The storage configuration determines how many copies of the file will be stored, where the copies will be stored, the checksum algorithm to use, and how the file should be grouped and packaged when sent to the storage device.

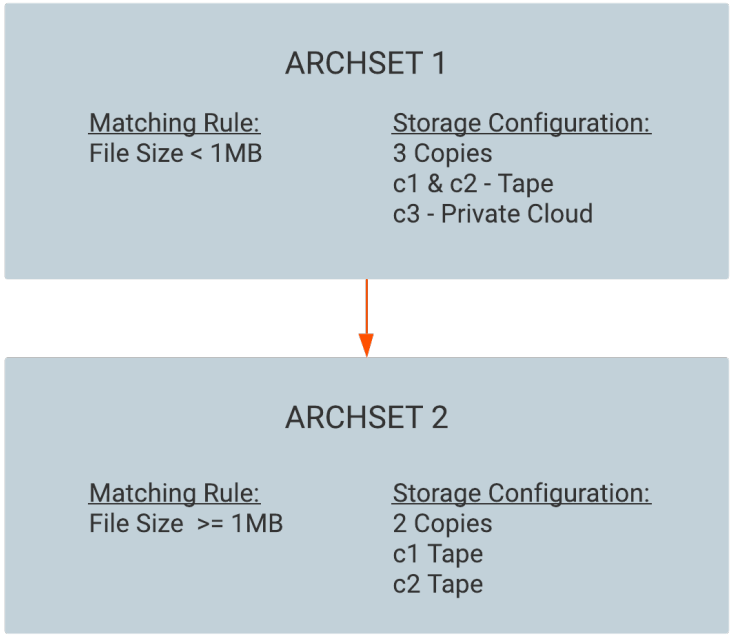
Together, a matching rule and storage configuration form a policy (also called an Archive Set or ARCHSET) in ScoutAM. An example of an ARCHSET definition for group matching is shown below.



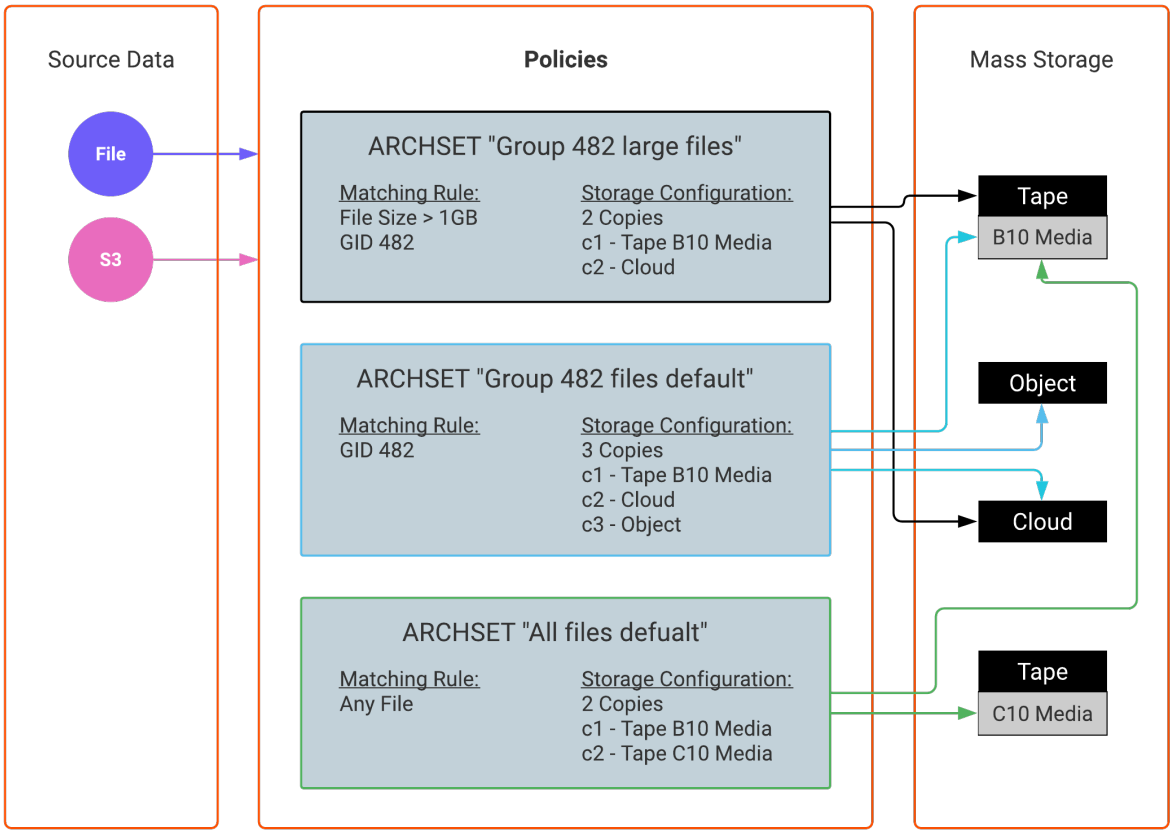
In this example the ARCHSET is named "Group 482 large files" and specifies a matching rule for files greater than 1GB that belong to a group with a group id (GID) of 482. It has a storage configuration for storing 2 copies, one on the storage target Tape B10 Media and one on the cloud.

The policy engine utilizes a first fit graphing algorithm to match against policies in a prioritized order, so a file may match multiple policies but only the first match is used to store the file. This means the most specific policies should be matched against first, and it is best practice to have a 'catch all' as the last policy so that any unmatched files are archived.

In the next example of an archset, the policy engine places files into groups based on size. All files have two copies archived to tape. In addition, a third copy is made for small files less than 1MB. This copy is made on a local object store. The system is configured so that small files are retrieved from the local object store, resulting in faster time for retrieval when those files are needed.



The policy engine can easily match on multiple different characteristics. For example, consider an organization that mandates that all files need to have two copies archived with at least one on tape, but the group with GID 482 wants all of their files less than 1GB available in an on-premises object store. This can be done with two policies that match files for the group with GID 482, plus a catch-all for the entire organization. Files for the group with GID 482 that are over 1GB will match the first policy and only have two copies stored, one on tape and one on the cloud, whereas the rest of the files for group 482 will match the more general rule that includes a third copy stored on an object store. Any file that remains unmatched at that point will match the last policy and will have 2 copies saved on tape.



An organization may have as many policies as necessary, with rules and configurations of varying complexity. Policies can be added, edited, and removed by the system administrator at any time, allowing ScoutAM to intelligently evolve with changing storage requirements.

Intelligent utilization of hybrid storage devices

The ScoutAM policy engine enables an organization to intelligently utilize their storage resources by taking advantage of the cost and performance characteristics of different storage devices while ensuring the data is not tied to specific vendors. For instance, an archive configuration that automatically copies data to both tape and object storage provides multiple characteristics, such as the ability to have data with an ‘air gap’ for security and also have fast access time to first byte for users since ScoutAM can be configured to access the copy on object storage first, and only revert to tape copies in the event of a read failure.

Sometimes separating specific data onto various types of storage media can dramatically speed up workflows. For example, some enterprise backup software solutions store small catalog files and large data files together. On serial devices like tape this can pose a major problem since the small catalog files provide metadata information which needs to be accessed often. Using just 2 simple policies, ScoutAM can automatically and transparently store small files on disk or object storage for quick random access, and large data files on tape, dramatically decreasing wait times for end users and lowering the number of tape mounts without changing anything from the user or backup software point of view. Both small and large files can be protected with secondary copies on tape.

Traffic Shaping For High Throughput

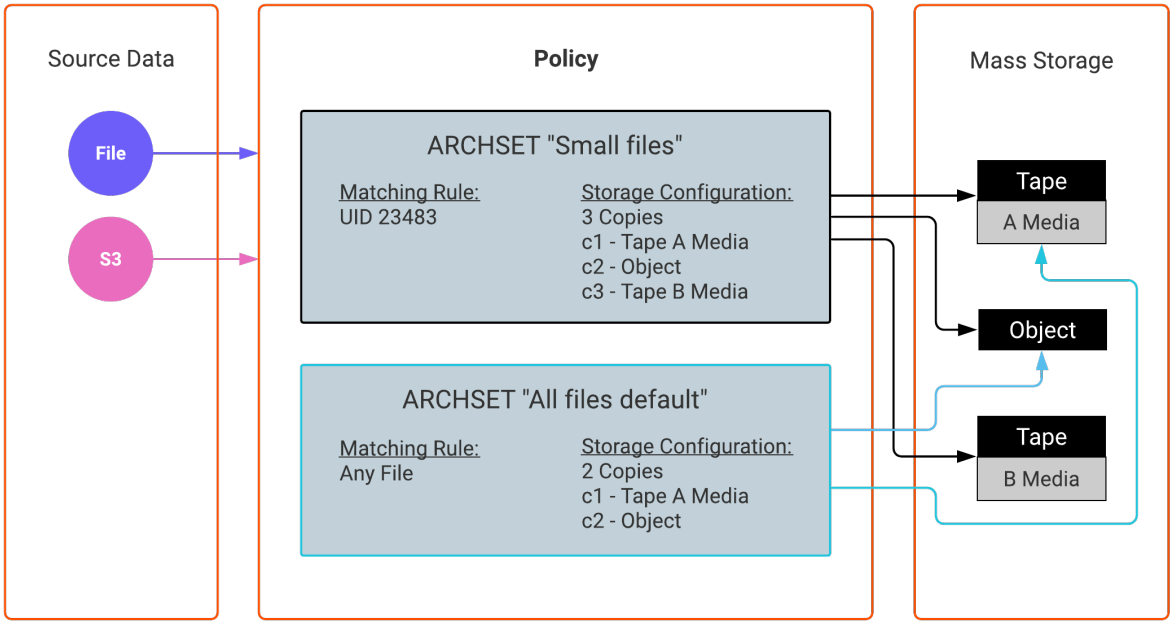
ScoutAM uses the policy engine to group files in order to achieve optimal streaming data rates to tape and object storage devices. Write performance of both tape and object storage systems are highly sensitive to the size of a given request. To obtain anything close to the rated throughput performance, large steaming file workloads must be used. However, most enterprise workloads contain a mixture of small and large files. Without traffic shaping, the small files can bring the systems to a crawl. ScoutAM addresses the need for traffic shaping by grouping files into GNUTar containers and streaming the containers to archival devices in a single streaming transaction. The ARCHSET definitions allow the administrator to set minimum and maximum container or group sizes. Controlling the workload in this manner allows ScoutAM to achieve very high throughput rates for archiving even in the presence of small files.

Seamless data migration

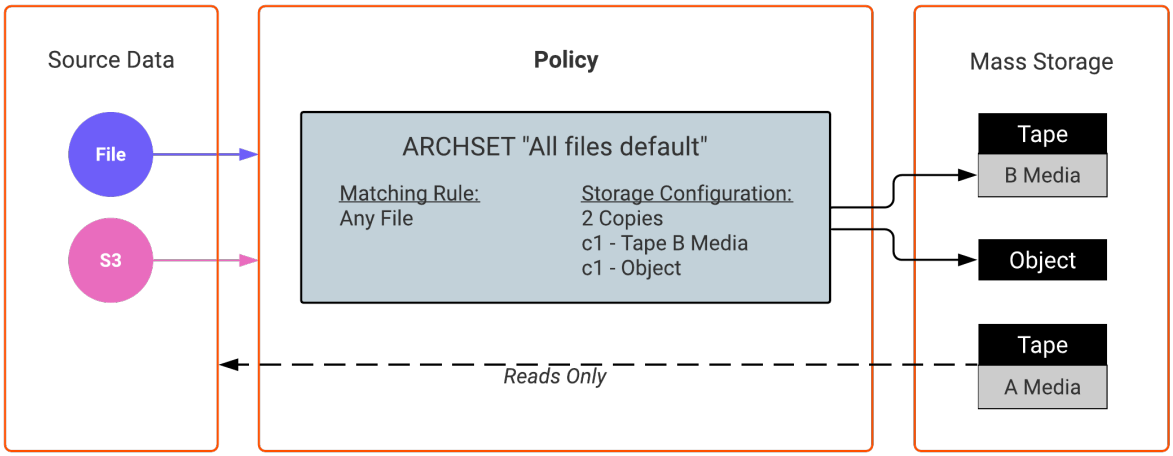
Because the data is decoupled from underlying storage hardware, ScoutAM allows organizations to easily adopt new types of archival storage as they become available. ScoutAM can automatically migrate data between different types of storage and hardware. Migration can occur over time in the background, providing minimal interruption to production workloads.

In ScoutAM the policy engine is used to determine where data is stored based on the configured policies for an organization, so data migration is easily performed by modifying or adding policies to redirect data. Since policies are used to perform data migration the power and flexibility of ScoutAM's policy engine can be used to roll out migrations for only specific sets of data or to perform a phased rollout. For example, an organization has decided to purchase a tape library from a new vendor and wants to do a phased rollout to the new hardware in order to gain experience with it before rolling it out across the organization. The decision is made to first test it out with a friendly user where they can direct additional copies of their files to the new

tape library and run tests to ensure it is working as expected. The system administrator simply adds one new policy in ScoutAM directing an additional copy of the files belonging to the friendly user (with UID 23483) to the new tape library (Tape Library B), while the rest of the organization continues to use the existing tape library (Tape Library A) and object store as shown below.



After gaining experience with the new tape library (Tape Library B Media Pool) the decision is made to move the entire organization to it and eventually decommission the old tape library (Tape Library A Media Pool). The system administrator only needs to change one line in the policy, directing data from the old tape library to the new tape library as shown below. At this point the policy for the friendly user can be removed as well. Note that the old tape library does not need to be in the policy configuration at this stage. If the organization had instead decided to augment their archival storage by adding a new tape library instead of decommissioning the old one they would be finished at this point. ScoutAM would continue to read files that already exist from the old tape library and would write all new files to the new tape library.



At this point the organization is ready to begin decommissioning the old tape library. This just requires running a single operation to re-evaluate the new policy on all existing files to move them to the new tape library. The system administrator simply issues a 'rearchive' or 'unarchive' command. The 'rearchive' command allows ScoutAM to continue to be able to read the copy from the old tape library before the new copy is created, this is useful in the case where a certain number of copies need to be maintained at all times, while the 'unarchive' command prevents any more use of the copy on the old tape library, useful in the case where the copy is on damaged or decommissioned hardware.

Conclusion

ScoutAM's intelligent archiving enables organizations to automatically and efficiently implement long term data preservation strategies. ScoutAM's rich policy engine automates the process of filtering and grouping data, creating copies, and moving data to any combination of mass storage devices. The ability to control the orchestration of data management policies enables organizations to continually optimize, cost, performance and accessibility for different kinds of data and evolving workloads.