A Novel Approach to Automated, Secure, Reliable, & Distributed Backup of MER Tactical Data on Clouds

George Chang and Khawaja Shams
John Callas, Alex Kern
Jet Propulsion Laboratory
4800 Oak Grove Dr.
Pasadena, CA 91109
(818) 393-4979
cloud@jpl.nasa.gov

Abstract— **In 2010, Mars Exploration Rover (MER) Project became the first NASA mission to incorporate a public cloud into its daily mission-critical operations. Since then, the operators and scientists have experienced 100% availability on tactical plan saves, searches, and retrieval. MER has also pushed tactical data onto lower cost storage systems in the cloud environments, which continue to deliver cost savings, durability, and reliability. Our architecture, designed for high availability and graceful degradation, has raised the confidence in cloud computing. The next step in MER's journey to realize benefits from cloud computing is to incorporate next generation storage solutions into data backup and retention strategy. The Operations Storage System (OSS) on MER currently contains 21 TB of data, which includes both uplink and downlink data. The data is currently backed up on secondary and tertiary devices internally, with some geographical redundancy by keeping snapshots in Simi Valley, less than a hundred miles away from JPL. Storage solutions offered as cloud services include high availability, geographical redundancy, extensive durability, as well as fine grained access control mechanisms. Through novel encryption techniques, data are uploaded without ever introducing the key into the cloud environment. We discuss our approach to optimize the various parameters, minimum requirements, and the challenges we faced during implementation and deployment. Our current design backs up data on S3 (Simple Storage Service). We evaluate the implications of our design, development costs, and the realized savings for the mission. We share performance and data retention benchmarks, and we conclude with an outlook on applicability of our process and application on other missions.**

TABLE OF CONTENTS

1. INTRODUCTION

Cloud computing continues to challenge traditional Information Technology (IT) and is driving it towards continued advancement. Over the last three years, several missions at the NASA Jet Propulsion Laboratory (JPL) have begun exploring the cloud, and they are already enjoying unprecedented computational capabilities in a cost-effective, secure, and streamlined fashion. Much of the research on cloud computing and its applicability to science applications focuses on the promise it holds for virtually limitless computational capacity. Although storage capabilities in the cloud also offer significant improvements over its traditional counterparts, it continues to be overlooked. The Data Services team at the NASA Jet Propulsion Laboratory has been working closely with Office of the CIO and mission management to help address complex challenges and leverage this underutilized cloud capability through novel storage techniques developed specifically for NASA applications.

Tape remains the state of the art for data backup across many organizations including JPL. Despite the ubiquity and proven reliability of tape archives, it has many well-known limitations. The slow recovery speed has been a major concern for missions. Meanwhile, some organizations still burn data on optical storage and ship it off-site for geographical redundancy. Cloud storage offers a compelling avenue for backups as it provides geographical redundancy, optimized recovery, high availability, and extremely high durability. Furthermore, major cloud computing vendors including Amazon Web Services, Google Cloud Storage, Microsoft Azure, and Rackspace have all started to offer free inbound traffic and only charge for outbound traffic. This pricing paradigm suits backups exceptionally well as it favors data coming into the cloud. Except in deeply unfortunate circumstances, backup applications employ a write-heavy pattern with reads limited to scenarios where data recovery is required.

2. CLOUD COMPUTING FOR BACKUP STORAGE

This section evaluates the efficacy of cloud computing as a viable solution for backup storage. It lists characteristics of a sound backup strategy and analyzes how a cloud computing solution may adhere to some of the best practices. Thematically, the three dimensions we try to optimize are durability, availability, and cost. Durability is a measure of how frequently data is lost. The main goal of any backup system is to increase the durability as much as possible. Availability measures the reliability of the data

store and the likelihood with which the data will be available when it is needed. A service outage for 4 hours, for instance, without any loss of data is considered a hit on availability but preserves durability. Cost is the limiting factor: our goal is to obtain as much durability as possible with minimal cost. Specifically, our goal is to significantly reduce cost compared to the traditional IT solution while enhancing the durability and availability of our data. Compliance, privacy and integrity of the data are considered high priority requirements, and any solution that even slightly compromises these requirements is deemed unsuitable.

Geographical Redundancy

Having offsite copies of data is crucial in ensuring long-term durability. Having data further away from the primary location geographically increases the level of durability by reducing the impact of regional outages. Cloud computing offers an institution the ability to replicate data in a remote location of its choice. For instance, Amazon Web Services allows users to chose between storage in Virginia, Northern California, and Oregon. Data stored in a metropolitan region remains in the region and is not transferred or replicated to other regions or countries. Cloud services also offer the ability to store data internationally, which is suitable for extremely high durability requirements. We chose to maintain all of our data, albeit encrypted, within the United States. Within each metropolitan region, there are several data centers where the data are replicated. For instance, to ensure the highest durability for the data, Windows Azure geo-replication makes copies of Azure Blobs and Azure Tables data across two data centers that are at least 100 miles apart at no additional cost. [1] Similarly, Amazon S3 (Simple Storage Service) offers a standard class of storage that can sustain the complete and simultaneous failure of two data centers. In fact, S3 is designed for 11 9's (99.999999999%/year) of durability for data. [2]

Cloud computing not only offers geographical disparity from the originating organization, but it also provides redundancy within its environments by replication of data across multiple data centers. Furthermore, this cross-data center redundancy within the cloud environment comes without any additional cost, complexity, or development. These characteristics make cloud computing a viable solution from the perspective of geographical redundancy.

High Availability

Imagine a scenario in which an organization loses a large amount of data due to a major catastrophe. In such cases, immediate restoration of the latest snapshots is of the essence to minimize disruption of business. In this situation, if the backup system is down or out for a while, it could result in severe exacerbation. Hence, it is crucial to choose a backup storage system that offers high availability. Recall that objects stored in the cloud environments offer automatic replication of data across multiple data centers. Aside from having a positive impact on durability, this architecture also offers a significantly higher availability. In case of an entire data center suffering from a network connectivity issue or outages, data retrieval requests are automatically routed to the replicated copy across the remaining data centers. This design helps us ensure that we can retrieve mission crucial data from the cloud as quickly as it is needed, even in case of simultaneous loss locally and a cloud data center outage.

Cost

One of the primary reasons organizations approach the cloud is because of the variable pricing model. The cost is directly correlated to amount of resources used and the level of service required. For storage services, the market has settled on comparable pricing schemes. Typically, ingress, or data upload is not charged. For storage, the leading providers charge the following:

- Amazon S3 (West Coast): <15.4 cents per GB/month, <12 cents GB egress
- Microsoft Azure: 14 cents per GB/month, 15 cents per GB egress
- Google Cloud Storage: <13 cents per GB/month, <12 cents per GB egress
- Rackspace Cloud Storage: 15 cents per GB/ month, 18 cents per GB egress

The market appears to have normalized on the pricing for now. In the case of Amazon S3, there are even more granularity in pricing, for example, data storage in different regions has different costs (West Coast is the most expensive). Amazon also offers reduced redundancy for even lower pricing. Because of the competitiveness of this market, we expect that prices will be even lower in the future and more storage options may also be offered.

Automation and Auditability

Automation is an absolute requirement in any backup system. State-of-the-art backup solutions have addressed this issue extremely well. However, automation comes at a cost. There are documented instances where extreme automation has led to oversight in the past. For example, if the backup daemon dies, it is possible for system administrators to not notice it until there is a need to recover the data. Furthermore, although the data may be written to the storage system regularly, there are no guarantees that it can be read back.

These problems can be easily addressed through an auditing process. In designing our backup application, we ensured that an auditing capability is a central component of our approach to validate that 1) backups are being done regularly and 2) retrieval is possible and any file corruptions should be addressed before data is lost. Fortunately, the storage systems in the cloud computing environments offer programmatic interfaces that allow us to build automation logic around our backup scheme. As detailed below, every day, our system tries to recover a random Martian day's

worth (sol) of data for MER and compares it against the source file system to ensure consistency. Furthermore, our design for automation and auditability encompasses a distributed approach, which ensures that even in case of failures of individual machines, we continuously audit and backup our data on a regular basis.

Performance

The backup and recovery process needs to be performed within a reasonable amount of time. Obviously, the backups must be able to keep up with the amount of new data collection. The recovery procedure must be able to fully recover data swiftly to minimize downtime suffered by the users. With the MER project, there are strict requirements to the amount of time allowed for recovery. The process must be able to recover the last 10 Martian days (sols) worth of data within 30 minutes of recovery initiation and also be able to recover 90 sols worth of data within two hours. These performance requirements can be met with more expensive backup systems. The cloud backup strategy allows us to offer this level of service at a very competitive rate.

Compliance, Privacy, and Integrity

Using a public cloud for data storage requires us to store data in data centers owned and managed by external entities. While much of the risks can and have been mitigated through understanding the characteristics of the new approach and working closely with the vendors, the best practices require us to minimize our exposure as much as reasonably possible. In our novel backup approach, all data stored in the cloud environment is compressed then encrypted with AES-128 before leaving JPL. Since encryption speed is not a bottleneck, it would be a trivial change for us to employ AES-256. However, the latest NIST specifications indicate that AES-128 is sufficient for the class of data we are backing up in the cloud. [3] It is crucial to note that the encryption key for the cloud is never introduced in the cloud. Our approach enables us to ensure the privacy as well as integrity of our data, while enjoying the aforementioned benefits available in the cloud. Since the key consists of a trivial amount of data, otherwise cost-prohibitive approaches, like tape, can be employed to backup copies of the keys.

3. METHODOLOGY

The process described in this paper can be decomposed into several discrete steps, all of which is orchestrated by a workflow system to allow for automated backups, audit, and recovery.

Server Setup and Data Structure

The server used to run the automated backup process uses modest hardware. It runs a quad-core Intel Xeon X5570 and has 8 GB of RAM. It is configured with a network-attached storage running on standard gigabit Ethernet which stores mission data and derived products. The data is organized into folders named after the sol, or Martian day, that it was collected. Each sol folder is distinct from one another with no overlapping data. Within each sol directory, there is a large directory tree with multitudes of files.

Backup

The backup process consists of four discrete steps, the archiving, compressing, encrypting, and transferring of data, as illustrated in Figure 1. For simplicity and clarity, each part of the process is isolated and explained in detail. In practice though, all four steps are combined into one continuous data flow. This continuity allows for better utilization of resources as each segment processes data as soon as it becomes available.

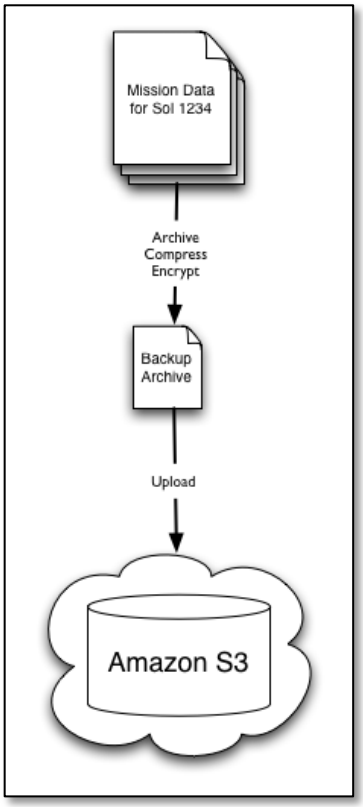

Figure 1 Backup Process

Archive—The first issue we had to consider was whether to store the backup as a mirror of the data, namely, keep each file as a discrete object in Amazon S3 or to bundle everything into one file using a tool like *tar*. We opted for the latter because the data typically has a large number of small files. Due to the nature of the data, the overhead of managing those small files across the network would have a severe performance impact.

The root directory of the archive is the top-level sol directory. This ensures that all data related to that particular sol is collected in just one file.

Compression—Because of the variable cost structure of using cloud storage -- we pay for exactly what we use -- it was in our best interest to reduce the amount of data that we transferred across the network as well as the size of the archive files stored.

The data that we were archiving is a heterogeneous mix of image data as well as science and planning information. We tried several compression algorithms and implementations, namely *zip*, *gzip*, *bzip2*, as well as threaded implementations of the *gzip* (*pigz*) and *bzip2* (*pbzip2*), to find the one that offered the best tradeoff between compression ratio and speed. All compression programs were used in their default settings.

Encryption—The data that we archive include the most recent data received from the rovers. As such, much of the data is ITAR sensitive and cannot be stored in decrypted format. We implemented encryption with OpenSSL with AES-128, satisfying government standards for encryption [3]. The encryption was applied on the compressed archive produced from the previous step. Upgrading to AES-256 is a simple parameter change in our software, and it consumes a trivial amount of additional resources.

Transfer—Once the data is encrypted, the resulting file is transferred to the cloud. Since we are specifically using Amazon's S3 storage, we are required to use Amazon's API for transferring data. We implemented this using a simple method of having just one open connection between the local machine and S3. In addition, we have developed a heavily optimized S3 client that employs multi-part uploads and downloads. By implementing multi-part uploads and downloads, we were able to leverage parallel processing and transfers to effectively use all available resources, both CPU and bandwidth to dramatically increase performance speeds.

Audits

The audit process is a daily process that randomly picks a saved archive on S3 and verifies that it still matches the data on the secondary backup. We do this to ensure the integrity of the backup archive.

The process to run the audit is effectively the reverse of the upload process with two main exceptions. The first is that the downloaded archive is processed in memory and nothing is written to local storage. Because we are just verifying the data, there is no need to save the data to local storage and incur performance degradations from more I/O processing. Secondly, after the data has been unencrypted and decompressed, we process each file in the tar archive in memory and generate an MD5 digest for each file. Concurrently, we traverse the local backup directory that corresponds to the backup archive and generate MD5 signatures. The two lists of MD5 signatures are then compared with each other to determine what files are the same and what files, if any, have changed. The output of this audit is emailed to the interested parties for further action, if necessary.

Recovery

The recovery process is a manual process run whenever the backup archive needs to be restored. Like the audit process, recovery is the inverse of the backup process. The backup archive is downloaded from S3, unencrypted, decompressed, and un-archived into the designated location.

Workflow and Usage

We used the Polyphony workflow system to automate the backup and audit process. Polyphony allows us to distribute the backup process across different machines, which we used to archive existing data. It also allows us to robustly schedule daily backup jobs and audit jobs, notifying us of any errors.

The output of each backup run and audit run is emailed to the interested parties with details about the number of files that have been backed up. For the audit report, any discrepancies between the local copy, if still available, and archive are noted and another backup process starts to refresh the cloud backup.

4. RESULTS

Backup

A random sol directory was picked as the dataset to benchmark performance. The directory contained almost 1.3 GB of data. Figure 2 shows the results of the backup process with different configurations.

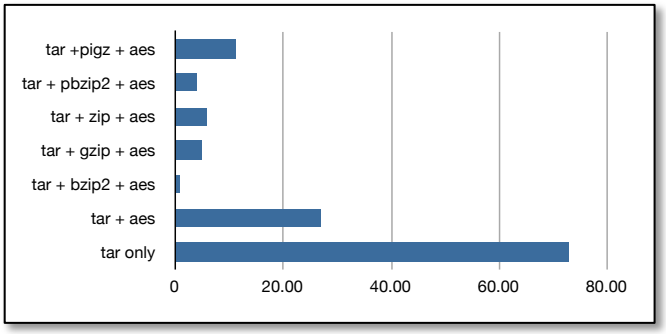

Figure 2 Backup Speeds (MB/s)

With a single upload connection, we experienced maximum speeds of approximately 5 MB/s. Using parallel transfers, we were able to concurrently upload 130 streams at the same time. As shown in the chart with the *tar* only configuration, this yielded performance of over 70 MB/s to Amazon's West Coast data center when just transferring raw data without any compression nor encryption. When encryption was applied, the performance was reduced dramatically from over 70 MB/s to about 25 MB/s. When compression was applied, the throughput was reduced even more.

The single-threaded implementations of the compression algorithms had the poorest performance while the multi-

threaded implementations had significantly better transfer speeds with *pigz* having the best performance.

In terms of resulting files sizes, Figure 3 lists the sizes of the archives using various compression schemes.

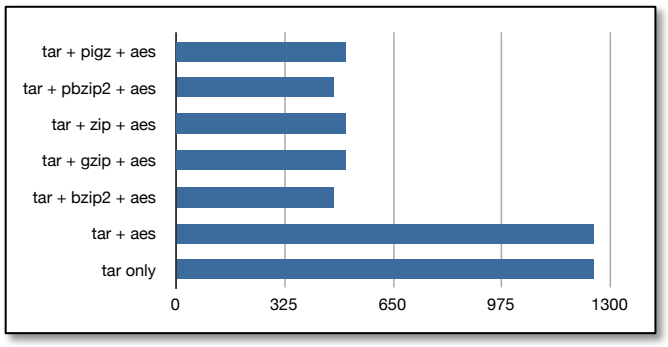

Figure 3 Archive Sizes (MB)

Without compression, the archive size is just under 1.3 GB. AES encryption, as expected, does not alter the size of the resulting file. When compression is applied, the savings were typically greater than 50%. In this particular case, any compression yielded a file size that was 40% of the originally size. The two implementation of bzip2 compression yielded smallest files.

Audit and Recovery

Given the results of backup process, we chose to use the *gzip* algorithm. The recovery procedure is the reverse of backup process. The file is transferred from Amazon S3, unencrypted, uncompressed (using single threaded *gzip* since *pigz* cannot perform parallelized decompression), and unarchived. The results for our dataset are shown in Figure 4.

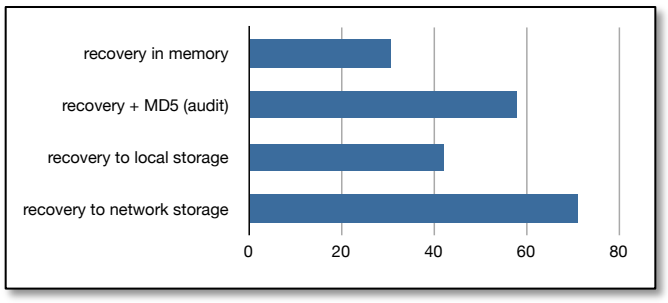

Figure 4 Time For Recovery and Audit (seconds)

Recovery into memory performs all the processing needed to download, decrypt, uncompress, and expand the archive but only into memory. This gives a clear picture of how long the process would take without local I/O constraints. For the 500 MB archive file, this took approximately 30 seconds, or 16 MB/s.

Recovery with MD5 measures the performance of the audit process, which performs the four steps to recover data as well as computing the MD5 digest of the files in the archive as well the MD5 digest of the files on the local backup. Because the MD5 digests are computed concurrently for both the archive and the local backup, there is some speedup. However, the entire audit process for the 500 MB archive file and the 1.3 GB local backup took approximately 1 minute.

Finally, the recovery to storage is to perform a full restoration of the files in the archive. We performed two restorations; the first restored the files to its original location, on the network-attached storage, and the second to a hard drive that is directly attached to the machine. With the local hard drive, we were able to recover the 1.3G of data in just over 40 seconds, while recovery to the network-attached storage took over a minute.

5. INTERPRETATIONS

The results show that our process has extremely high performance for backups and recovery. Our multi-threaded transfer client was able to give us transfer rates that far exceed common means archiving data:

- Parallel transfers to S3: 70MB/s sustained
- External USB hard drives: 60MB/s sustained (theoretical based on USB specifications)
- DVD writer at 24x: 31 MB/s sustained (theoretical)

In addition to the superior transfer speeds compared to other mediums, our process also has several other distinct advantages. The addition of compression and encryption makes the entire process very seamless and can be run as a daily job that performs the backups in the background. The cloud also gives us virtually unlimited storage without needing reconfiguration, so we do not need to worry about running out of space as we would with discrete hard drives and DVD discs. The audit process also allows us to constantly test the integrity of the backup to ensure that we can do a full recovery when we need to.

Despite the large amount of data that the mission produces everyday, our backup process is able to keep up with archiving. Given an average of 1 GB compressed archive per sol and the speeds obtained with encryption, parallel compression, and transfers, we can archive three Martian months worth of data in approximately three hours.

Recovery Performance

In order to meet our requirements, we need to show that we're able to recover 10 sols in 30 minutes and 90 sols in 2 hours. Our tests show that the first requirement can be met by restoring to both network-attached storage as well as local disk. The second requirement can be satisfied when writing to local disk. This result is quite surprising. Namely, the bottleneck in utilizing cloud storage is not the transfer

rate to the cloud servers hundreds or thousands of miles away, but rather, it is within our own hardware systems.

Costs

With the current cost model for Amazon S3 storage, the price for archiving data is relatively inexpensive for the reliability and performance. Currently, all upload transfers are free, so there is no cost for moving data to the cloud. To store one sol's worth of data, or approximately 1 GB on average, the monthly cost would be at most 15 cents/month with a declining scale as data accumulates. To perform an audit or a recovery would cost at most 12 cents, also with declining prices as data transfers increase.

6. Conclusions and Recommendations

Our backup process to the cloud, along with the variable cost model of cloud storage, provides a compelling alternative to traditional backup processes. The optimized data transfer client can support speeds that exceed common data backup strategies such as using an external hard drive and DVD archiving. The seamless integration of archiving, compressing, and encryption, along with virtually unlimited storage, adds to the convenience of our backup process. The automated auditing ensures that backups are reliable and up to date and when recovery is needed, data can be restored in a very short amount of time.

Further areas of study can focus on increasing even more performance. Our current setup runs on very modest machines with very few tweaks in compression and encryption parameters. Distributing backups between different cloud vendors could also be another area of improvement. Although our software is modular, we are currently only employing Amazon S3 as the backup data store. If data are distributed across different vendors, the backup process can be even more robust and tolerate not just failures in a particular geography, but rather, the entire vendor.

Acknowledgements

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] Microsoft Azure Web Site: http://www.microsoft.com/windowsazure/features/storage/

[2] Amazon S3 Web Site: http://aws.amazon.com/s3/

[3] CNSS Policy No. 15, Fact Sheet 1. National Policy on the Use of the AES Standard to Protect National Security Systems and National Security Information. http://csrc.nist.gov/groups/STM/cmvp/documents/CNSS15FS.pdf

BIOGRAPHIES

George Chang *is a Senior Software Engineer at NASA's Jet Propulsion Laboratory (JPL). He co-developed the backup and recovery process. He started working at JPL as a college intern and in his 8 years at JPL so far, he has been intimately involved in the data systems, security, and infrastructure of NASA's Deep Space Network and Lunar Mapping project, in addition to his work for MER. He constantly tries to infuse new advances in computer technology such as cloud computing and mobile devices into his projects. He has a B.S. in Computer Science from Cornell University, an M.S. from Columbia University, and has an M.B.A. from the Anderson School of Management at UCLA.*

Khawaja Shams *is a member of the Operations Planning Software (OPS) Lab at the NASA Jet Propulsion Laboratory. Khawaja develops software that contributes to the operations of a variety of robotic assets including ground, airborne, and waterborne robots, as well as robots on Mars. He leads a variety of software projects, and he serves as the Cognizant Engineer of server side components for the Activity Planning and Sequencing Subsystem (APSS) for the Mars Science Laboratory. Khawaja works closely with the Office of the CIO at JPL to co-lead the efforts to securely deliver the benefits of cloud computing to missions across NASA. He serves as an advisor on the CIO Technology Advisory Board (CTAB) at JPL. Khawaja is currently pursing a PhD in robotics at USC, holds a Masters in Computer Science from Cornell, and a Bachelors in Computer Science from UC San Diego.*

John L. Callas*, of NASA's Jet Propulsion Laboratory, Pasadena, Calif., has been project manager of NASA's Mars Exploration Rover project since March 2006. Previously, as science manager and then deputy project manager, he had helped lead the rover project since 2000. Callas grew up near Boston, Mass. He received his Bachelor's degree in Engineering from Tufts University, Medford, Mass., in 1981 and his Masters and Ph.D. in Physics from Brown University, Providence, R.I., in 1983 and 1987, respectively. He joined JPL to work on advanced spacecraft propulsion, which included such futuristic concepts as electric, nuclear and antimatter propulsion. In 1989 he began work supporting the exploration of Mars with the Mars Observer mission and has since worked on seven Mars missions. In addition to his Mars work, Callas is involved in the development of instrumentation for astrophysics and planetary science, and teaches mathematics at Pasadena City College as an adjunct faculty member.*

Alex Kern *currently attends Beverly Hills High School. From a young age, he was been interested in science and technology. He began programming when he was 9 and received his first paying website design job at 11. He has since interned at NASA, JPL for the past two summers working on novel data storage techniques. During the school year, Alex spends time managing his high school robotics team and working for an LA-based web startup. He wishes to become a technology entrepreneur.*