July 2010 NDIIPP Partners Meeting Breakout Session

Data Integrity: Monitoring in the Cloud
Breakout Session #10

NDIIPP Partners Meeting
Wednesday July 21, 2010
4:15 p.m. – 5:30p.m

Presenters: Mike Smorul, University of Maryland
Andrew Woods, DuraSpace

Attendees: 13

"Cloud Integrity Monitoring"

Mike Smorul, University of Maryland, presented "Cloud Integrity Monitoring." Mike is Lead Programmer at University of Maryland Institute for Advanced Computer Study and works on the ADAPT project which is "developing technologies for building salable and reliable infrastructures for the long-term access and preservation of digital assets" (https://wiki.umiacs.umd.edu/adapt/index.php/Main_Page).

Mike's presentation introduced ongoing work on an extension of ADAPT's ACE tool to cloud computing. This extension addresses a key issue in sharing resources in the cloud—how can a third party validate that shared content? Additionally, how can a third party do this without incurring additional transfer costs.

ACE (Auditing Control Environment) provides continuous auditing using cryptographic methods. It comprises two main components, an Audit Manager and an Integrity Management Service (IMS), as well as a GUI. Two data units in ACE validation are the digest string (MD5, etc.) and the IMS-issued token for each file/hash that is to be verified. This token includes a cryptographic proof or Cryptographic Summary Information (CSI), a number that can be stored externally to the system.

Mike described the several scenarios for validation in the cloud using ACE. In one scenario a third party downloads an object and runs validation processes using external information. That is, the user acquires the token and the original file, computes the digest for the file, computes the "CSI" values using the token and the digest, and finally compares the newly generated CSI values to the remote CSI values on the IMS (which is public and not tied to the depositor). Another scenario would implement validation during processing, uploading the routines with ACE which would then compute the digest during access and read processes.

Discussion:
The attendees wanted to know when ACE application for cloud computing would be available? Soon.
There followed a brief discussion of the issue of file size in the cloud.

“DuraCloud: Data Integrity Monitoring in the Cloud”

Andrew Woods, a lead developer for DuraCloud, spoke on data integrity services in the cloud. DuraCloud is a service application, not the storage itself. Its focus is on supporting preservation activities in the cloud and data access for use and sharing. The service is designed to be a kind of “open canvas for cloud-based services.”

DuraCloud currently services 30 TBs and Boston public TV, WGBH, and the New York Public Library are among its clients. A java client wraps REST calls and at a higher level a sync tool collates source and cloud data. Content is accepted via http or hard drives. Md5 hash values are generated in transfer and also after content lands in the cloud. Alternatively clients can provide their own md5 values and DuraCloud will process and compare those.

Woods described several approaches to verifying content provided by DuraCloud:

- request stored value and perform verification outside the cloud (inexpensive and fast)
- stream out content and recalculate (compute intensive and slow)
- stream out content and recalculate with “salt” (user intensive, compute intensive, and slow).

Verifying with salt involves appending a string to the data object and generating entirely new md5 values and proceeding from there. The objective is to make a “new thing” that the verification system is forced to recognize and process. This is a form of verification on the storage system process itself and assures the user that the system is not providing a false report for data, especially if it has been held in the cloud for a long period of time.

Woods closed by describing “next steps” for DuraCloud verification processes. These include addressing scalability issues and implementing event logging.

Discussion:
The audience asked for clarification on verifying with salt, which was provided as above.

Where do the actual md5 values reside? With the data object. DuraCloud is agnostic on this matter. It’s up to the storage system provider.

Where will event logging output be stored? Currently on the computing system but ultimately may push it to another location. These are not large files current solution requires paying for it in the store. Are the event logs associated with the objects? No.

Is generating METS for an object an event? No. DuraCloud enables interaction with storage and services. It currently does not update files but instead supports a business logic that you invoke yourself.

DuraCloud services then would not do a fix on files? No. It just flags and reports. The user has to do the forensics and do something about the problem file.

There was a general discussion and explanation provided by Mike Smorul and Joseph F. JaJa (University of Maryland) of different hash algorithms. The stronger the hash, the more secure it is, but also the more intensive the compute power required to process it. But none of the hashes provide 100% security. All of them, ultimately, may be hacked.

There was a brief discussion about how to use DuraCloud generally. It has three Web apps for storage, service, and administration and provides an interface that runs in the browser. The user gets an account. You can download, build, and install locally in your own tomcat, but it’s just easier to get an account and use the remote service. The fees for the DuraCloud instance and the fixity service are separate.

The session closed with a brief discussion of possibilities and issues in implementing ACE in DuraCloud. Both presenters expressed interest in exploring this.