DSHR's Blog

I'm David Rosenthal, and this is a place to discuss the work I'm doing in Digital Preservation.

Tuesday, September 3, 2013

Talk for "RDF Vocabulary Preservation" at iPres2013

The group planning a session on "RDF Vocabulary Preservation" at iPRES2013 asked me to give a brief presentation on the principles behind the LOCKSS technology. Below the fold is an edited text with links to the sources.

The two most important questions to ask when designing a digital preservation system are:

- What digital information am I supposed to preserve? The bane of digital preservation has been the idea that somewhere out there a one-size-fits-all system can be found. The LOCKSS Program published our answer to this question in 2000 (PDF). We designed the system to preserve information, primarily copyright material such as academic journals, published on the Web. Limiting our ambition in this way made it feasible to build a complete, working system rather than a demo.
- What am I supposed to preserve the digital information against? Or in technical jargon, "what is the threat model?" We published our answer to the second question in 2005, as a list of the causes of data loss we thought significant, and how we mitigated them.

If you ask most people for their list, you would get something like this:

- Media failure
- Hardware failure
- Software failure
- Network failure
- Obsolescence
- Natural Disaster

It's true, all these are causes of data loss. But if you ask the people who run large data centers what are the most important causes of data loss, you get a list like this:

- Operator error
- External Attack
- Insider Attack
- Economic Failure
- Organization Failure

The LOCKSS system was explicitly modelled on the paper library system, for a number of reasons:

- The customers were paper libraries; we wanted a system they could understand.
- Paper libraries were used to handling copyright material.Working around the copyright law was our single most important design goal.
- Paper libraries had evolved over millennia into a remarkably effective system for preserving information.

The paper library system created lots of copies of material to be preserved, on durable, somewhat tamper-evident media, and scattered them around the world in such a way as to make it easy to find (and potentially alter or destroy) some of them but hard to be sure that you had found all of them.

The only part of this that wasn't practical to emulate in the digital world was the durable, tamper-evident media. We had to make up for that with technology. The way the system

DSHR

DSHR in ANWR

My Projects

- LOCKSS (a trademark of Stanford University)
- CLOCKSS

LOCKSS system has permission to collect, preserve, and serve this Archival Unit.

Blog Archive

works is simple in essence:

- The publisher grants permission on their Web site, either via a Creative Commons license, or by an explicit statement that LOCKSS has permission to collect and preserve it. For subscription content, this must be some place that only subscribers can see.
- Each LOCKSS box independently collects the content whose permission statement it can see, by crawling the publisher's Web site. This isn't a completely reliable process, but the last step fixes any discrepancies.
- The LOCKSS boxes can serve their content to readers via the Memento HTTP protocol, which provides seamless WayBack Machine-like access to preserved Web content. Memento allows content from a URI (such as a journal) that is preserved at some other URI (such as at the WayBack Machine) to be retrieved from the original URI, even if the original URI knows nothing about Memento.
- The LOCKSS boxes cooperate in a peer-to-peer network to detect and repair damage to their contents. The protocol they use to talk to each other is complex in detail as it has to defend against numerous possible attacks (PDF), but simple if you ignore these defenses. At intervals each box (the poller) creates a random sample of the other boxes (the voters) with the same content, and gets them to vote on its hash. If the poller agrees with the consensus of the voters, all is well. If not, it fetches a repair from one of the boxes which do agree with the consensus, or from the publisher if it still available.

To make this work, you need a minimum number of copies and thus a minimum number of LOCKSS boxes. We like to have at least 7 copies in order to start feeling confident that the material is safe. More boxes is better; the more you have the lower the probability that two boxes will share the same content, which improves resistance to attack. For example, the Global LOCKSS Network has about 150 boxes, and the median journal volume has of the order of 25 copies scattered among them.

Smaller, private LOCKSS networks (PLNs) also work. For example, the CLOCKSS PLN currently has 12 boxes. Each is configured to contain all the content preserved by the CLOCKSS Archive, so they all have the same content. The PLN takes other defensive measures to make up for this correlation.

Bringing up a physical or virtual LOCKSS box is simple, using a custom Linux Kickstart image. Running a network of boxes requires some management. Some PLNs do this for themselves, and others have it done for them by the LOCKSS team.

Preserving new content involves writing a "plugin" that describes the content to the system, including details such as where it is on the Web, what its boundaries are, when the box is allowed to crawl it, how to detect and filter out personalizations, and so on. Plugins are XML with, in some cases, embedded Java classes. You can have the LOCKSS team write one for you, or do it yourself.

There are even networks you can join, such as the MetaArchive, that arrange the whole preservation process for you.

Posted by David. at 8:43 AM
Labels: digital preservation, ipres2013, memento

No comments:

Post a Comment

Links to this post

Create a Link

Home Older Post

Subscribe to: Post Comments (Atom)

► 2012 (43)
► 2011 (40)
► 2010 (17)
► 2009 (8)
► 2008 (8)
► 2007 (14)