LOCKSS Auditing using the SAFE-Archive System

Prepared for PLN2010
Boston College, Boston, MA

Jonathan Crabtree
October 26, 2010
The Odum Institute

- Oldest Institute or Center at UNC-CH
  Founded 1924
- Mission: Teaching, research, & service for social sciences
- Cross-disciplinary focus
Collaborators


• Research Support
  Thanks to the IMLS (LG-05-09-0041-09) and to the Data-PASS member institutions for support of this research.
The Data-PASS Partners

- ICPSR
- Odum Institute
- Roper Center
- Henry A. Murray Research Archive
- Harvard-MIT Data Center
- National Archives and Records Administration
Replication

“Storage alone will not solve the problem of digital preservation. Academic materials have many enemies beyond natural bit rot: ideologies, governments, corporations, and inadequate budgets. It is essential that sound storage and administration practices are complemented with the institution of communities acting together to thwart attacks that are too strong or too extrinsic for such practices to protect against.”

(Maniatis et al. 2005, 41)
SAFE-Archive Project

- Funded under an IMLS award to IQSS, Harvard U.
- Open Source
- TRAC Based Audits
- Schema Based Policy
- Web-based Tools
- Adheres to LOCKSS Trust Model
SAFE-Archive Audit Server Components

• Red Hat LINUX Server
• Glassfish
• JAVA Server Faces
• MySQL
• Eclipse BIRT Web Viewer
• Custom Applications
Facilitating collaborative replication and preservation with technology…

- **Collaborators** declare explicit non-uniform resource commitments
- **Policy** records commitments, storage network properties
- **Storage layer** provides replication, integrity, freshness, versioning
- **SAFE-Archive software** provides monitoring, auditing, and provisioning
SAFE-Archive Design

• Central audit server collects network status
  – Information gathered via servlet calls to LOCKSS caches
  – Network status stored in local MySQL tables
• Audit uses XML schema in comparison with network status tables
  – Policies transformed into schema via user based questionnaire
  – Audit reports stored in both detailed XML file and report tables for presentation
  – Web-based interfaces provided for raw data tables & formatted reports
  – Audit reports are stored in secure file system along with policy schema used
• Solutions to audit problems automatically initiated by server
  – State of network is compared with policy requirements
  – Additional caches are instructed to harvest collections that are under-replicated
Program Design

- Five Central SAFE Subsystems
  - SAFE Schema Manager
  - SAFE LOCKSS Daemon Status Extractor
  - SAFE Comparison Tool
  - SAFE Web-based Reporting & Interfaces
  - SAFE Preservation Enforcer
SAFE LOCKSS Daemon Status Extractor

- Identify LOCKSS caches to be audited
  - User provided IP’s
  - Central Plug-in Registry & LOCKSS.XML
- User Provided Debug Account
- Http Servlet Calls to caches
- LOCKSS Caches return XML
- XML parsed into MySQL tables
Extractor Data Processing

- Iterations through data from various caches
- Required audit data stored in summary tables
- Automated cron jobs refresh data in user defined intervals
Extractor Inputs/Outputs

• Inputs
  – Identity of LOCKSS caches to be audited
    • Web-base user input of individual IP’s
    • Location of Plug-in registry & LOCKSS.XML

• Outputs
  – Audit Summary Tables in MySQL
  – Web-base Audit Table Viewer
    • SAFE LOCKSS Daemon Status Data Viewer
  – Provides input data for Schema Manager and Comparison tools
SAFE Schema Manager

- Web-based questionnaire user interface
- Policies expressed in human readable text
- Policies output into XML based schema
- Schema designed to be TRAC compliant
XML Based Questionnaire

<?xml version="1.0" encoding="UTF-8"?>
<questionnaire xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="QuestionnaireDefinition.xsd">
  <question>
    <questionText>What is the name of the private LOCKSS network (PLN) that you would like the SAFE system to audit?</questionText>
    <explanation>A private LOCKSS network (PLN) is a set of LOCKSS servers that are joined together to preserve your collections. Please select an existing PLN from the list or <!--LINK:-->define a new PLN</LINK>.</explanation>
    <xpath>//SSP/network/networkIdentity/name</xpath>
  </question>
  <question>
    <questionText>What is the email address of the PLN administrator?</questionText>
    <answerFormat>(\w+@\w+.\w+|)</answerFormat>
    <xpath>//SSP/network/networkIdentity/accessBase/@adminEmail</xpath>
    <errorText>You must enter a valid email address in this field!</errorText>
  </question>
  <selectionSet>
    <selector>
      <!-- Question text if there is an explanation tag for a selector element
      <questionText>Please select the servers you wish to audit.</questionText>
      <explanation>Above is the list of LOCKSS servers available to audit the were discovered. Please select the servers that to audit. If a server is missing, you should LINK:reconfigure the PLN</LINK>.</explanation>
      -->
Web-based user interface

Question: 0. What is the name of the private LOCKSS network (PLN) that you would like the SAFE system to audit?

Your answer: 

Explanation

Recommended answer:

Next >
SAFE-Archive Policy Schema

- **Network level:**
  - Identification: name; description; contact; access point URI
  - Capabilities: protocol version; number of replicates maintained; replication frequency; versioning/deletion support
  - Human readable documentation: restrictions on content that may be placed in the network; services guaranteed by the network; Virtual Organization policies relating to network maintenance

- **Host level**
  - Identification: name; description; contact; access point URI
  - Capabilities: protocol version; storage available
  - Human readable terms of use: Documentation of hardware, software and operating personnel in support of TRAC criteria

- **Archival unit level**
  - Identification: name; description; contact; access point URI
  - Attributes: update frequency, plugin required for harvesting, storage required
  - Terms of use: Required statement of content compliance with network terms; Dissemination terms and conditions

- **TRAC Integration**
  - A number of elements comprise documentation showing how the replication system itself supports relevant TRAC criteria
  - Other elements that may be used to include text, or reference external text that documents evidence of compliance with TRAC criteria.
  - Specific TRAC criteria are identified implicitly, can be explicitly identified with attributes
  - Schema documentation describes each elements relevance to TRAC, and mapping to particular TRAC criteria
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SSP xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="SSPSchema_v1_03.xsd">
    <audit audit_id="safe-audit">
        <schemeVersion/>
        <auditReportEmail>ed_bachmann@unc.edu</auditReportEmail>
        <auditReportInterval maxDays="1"/>
    </audit>
    <network>
        <networkIdentity>
            <name>safe</name>
            <accessBase adminEmail="jonathan_crabtree@unc.edu"/>
        </networkIdentity>
        <networkCapabilities/>
        <networkTerms/>
    </network>
    <hosts>
        <host host_id="152.2.32.205">
            <hostIdentity>
                <name>fong.irss.unc.edu</name>
            </hostIdentity>
            <hostCapabilities>
                <storageAvailable max_size="10"/>
            </hostCapabilities>
            <hostTerms/>
        </host>
    </hosts>
</SSP>
Schema Manager Inputs/Outputs

• Inputs
  – Web-base user policy selections
  – Network details from Extractor Tool

• Outputs
  – Schema.xml file
SAFE Comparison Tool

- Retrieves network status from tables
- Parses Schema.xml from Schema Manager
- Performs tag by tag comparisons
- Initiate unit conversions and data cleaning
- Prepare audit output
  - Generate diff.xml difference report
  - Create reporting tables
  - Store reports in secure file system
    - Schema.xml, diff.xml, birtreportdesign.xml, birtreport text/pdf
Current Policy Comparisons

• AU Level Comparisons
  – Number of Verified Copies “Number of copies with 100% agreement”
  – AU Size Commitment “Has the AU grown too large?”
  – Verification Frequency “Time since last 100% agreement”
  – Crawl Frequency “Time since last successful crawl”
  – Operational reporting
    • Crawl Duration “Amount of time during last crawl”

• Host Level Comparisons
  – Disk Space Commitment “Are the servers providing correct storage?”

• Network Level Comparisons
  – None
Future Policy Comparisons

• **AU Level Comparisons**
  – Geographic Distribution “Are the AU’s in distinct geographic regions?”
  – Operational reporting
    • Display Current Agreement less than 100%

• **Host Level Comparisons**
  – Geographic Distribution
  – Operational reporting
    • Health of Cache - Min Up Time, Max Unreachable Incidents, ETC

• **Network Level Comparisons**
  – Minimum Free Disk Space Policy
  – Minimum Number of Hosts Policy
  – Geographic Distribution Policy
SAFE Audit Reporting

- Item by item detailed XML comparison
- Customizable BIRT web-based reports
- Web-based searchable audit history reports
- Web-based table view of current raw network status data
<hosts>
  <host host="137.99.36.160" seq_num="1">
    <test>
      <schema_xpath>/SSP/hosts/host/@host_id</schema_xpath>
      <schema_value>137.99.36.160</schema_value>
      <db_column>ip_address</db_column>
      <db_value>137.99.36.160</db_value>
      <match>true</match>
    </test>
    <test>
      <schema_xpath>/SSP/hosts/host/hostCapabilities/storageAvailable/@max_size</schema_xpath>
      <schema_value>7516192768</schema_value>
      <db_column>repo_size</db_column>
      <db_value>1700000000000</db_value>
      <match>true</match>
    </test>
  </host>
</hosts>
Customizable BIRT Reports

- Eclipse based XML style report design
- Web-based BIRT viewer provides simple access
- Required audit report templates included
- Cron jobs automatically store reports in secure file system
- Users can run manual reports on demand
- Advanced users can customize reports
- Multi-format report output
  - PDF, Word, Excel, Postscript, Powerpoint
  - BIRT Reports
Audit Report Snapshots

- Regular audit reports stored in file system
- Web-based display of audit materials
- Search by audit date range to be added
- Included with report are the associated files used in comparison.
  - Schema.xml, Diff.xml, etc.
SAFE Preservation Enforcer

- Reads Schema.xml
- Reads Diff.xml audit report
- Determines policy “Red Flags”
- Determines provisioning solution
- Format query to LOCKSS cache
  - Request harvest of AU
  - Initiate a poll
  - Email an Administrator
  - Etc…
SAFE Archive Challenges

• Integrate tools in user friendly web interface
• Audit time synchronization
• Schema versioning/instances
• AU/Collection/Plug-in management
• AU scalability
• Bulk file restoration
• Hardware level policy enforcement
• Streamline SAFE Audit Server installation and distribution
• Integrate and streamline PLN & LOCKSS creation & installation
Current Development Schedule

• Exposing Content and Policies
  – [OCTOBER] Extensions to the Dataverse Network Systems
    -- easily enable curators to expose their content for LOCKSS harvesting
  – [Fall 2010] Version 1.0 of the Safe-Archive schema and related documentation

• Support for policy based auditing
  – [February 2011] Version 1.0 BETA of the SAFE-Archive system, 
    -- policy creation, and auditing of LOCKSS network
    -- including Amazon AMI-based installation

• Automatic Provisioning
  – [Fall 2011] Version 1.1
    -- supporting auto-reconfiguration of LOCKSS networks to reflect changes to policy schema

• Training
  – [Fall 2011] Online courses and written guide
Contact Information

data-pass.org

Jonathan Crabtree
jonathan_crabtree@unc.edu