

# The International Criminal Tribunal for Rwanda Information Heritage Project (aka Voices of the Rwanda Tribunal): Integrity Verification Architecture

## Lead System Architects

**Alexei Czeskis** University of Washington, [aczeskis@cs.washington.edu](mailto:aczeskis@cs.washington.edu)  
**Karl Koscher** University of Washington, [supersat@u.washington.edu](mailto:supersat@u.washington.edu)

## Lead Media Technologists

**Max Andrews** University of Washington, [madmax@nyu.edu](mailto:madmax@nyu.edu)  
**Nell Carden Grey** University of Washington, [ncgrey@u.washington.edu](mailto:ncgrey@u.washington.edu)

## Principle Investigators

**Batya Friedman** University of Washington, [batya@u.washington.edu](mailto:batya@u.washington.edu)  
**Tadayoshi Kohno** University of Washington, [yoshi@cs.washington.edu](mailto:yoshi@cs.washington.edu)

## Chair, Advisory Board

Donald Horowitz

## Additional Team Members

Patricia Boiko, John McKay, Lisa P. Nathan,  
Ronald Slye, Robert Utter, and Elizabeth Utter

UW Computer Science and Engineering Technical Report #09-01-02

# 1 Introduction

Following the 1994 Rwandan genocide, the United Nations established an international criminal tribunal (one of the first since Nuremberg in the 1950s) to prosecute those who masterminded the genocide. As of this writing, that tribunal was scheduled to start winding down in late 2008, and coming to a close in 2010. In order to collect the insights, knowledge, stories, and personal reflections of those who “lived and breathed” the tribunal, in the fall of 2008, Principal Investigator Batya Friedman (Professor, Information School, University of Washington) and Judge Donald Horowitz (former Washington State Superior Court Judge) composed a team of information scientists including Lisa Nathan, legal experts including John McKay (Professor, Seattle University School of Law and former U.S. Attorney), Eric Saltzman (founder of Creative Commons), Ronald Slye (Associate Professor, Seattle University School of Law), and Justice Robert Utter (former Chief Washington State Supreme Court Justice) as well as award-winning cinematographers including Max Andrews, Patricia Boiko, and Nell Carden Grey. The team traveled to Arusha, Tanzania (where the ICTR is located) and to the investigative arm of the tribunal in Kigali, Rwanda to conduct 49 video interviews with tribunal personnel. The interviews were brought back to the University of Washington with the intention to store and preserve them for future use.

Those digital video interviews constitute the core components of the corpus that we describe in this document.

**Digital Video Corpus.** The digital video corpus is characterized as follows: (1) 49 video interviews, each lasting approximately 1.5 hours; (2) interviewees include judges, prosecutors, defense counsel, interpreters, court administrators, investigators, the warden and other tribunal personnel; (3) approximately 5 terabytes of digital data; and (4) collected independently from the United Nations. We have also added to this corpus additional information and tools to facilitate the correct interpretation and authenticity verification of this corpus.

**Preserve, Repurpose, Reuse.** In collecting these materials, we were not acting as investigative journalists or as documentary film-makers. Rather, our intent was to act as a voice for the globe, to ask the questions and elicit the reflections and stories which others now and well into the future might wish to hear from those who enacted the tribunal. If we did our work well, then Rwandans seeking healing and reconciliation, legal scholars seeking to support justice system capacity building, journalists, documentary film-makers, high school teachers, and others could come to the corpus and find material to speak to their concerns and projects. Thus, our goal is to enable others to repurpose, reuse, and remix these materials. Furthermore, we want these materials to be widely available across multiple lifespans in a form for which the integrity of the original corpus and vetted extensions can be verified. Our general strategy for achieving this entails the use of cryptographic hash functions, widely disseminated and archived hash values, and the secure preservation of original materials.

## 2 The Original Corpus

### 2.1 Contents

The original corpus consists of 49 video interviews, supplementary videos of Arusha and Kigali, files describing the corpus file structure, public keys for cryptographic asymmetric signature schemes, reference software for verifying the integrity of the corpus, supporting standards, and this document.

In the corpus, this document is called `Integrity_Verification_Architecture.pdf`.

**Context and Videos.** There are several properties to note about this corpus. First, at the time of this writing, the entire corpus is not available for the public at large to download and access. There are two reasons for this. First, because of our wide-spread accessibility goals, we have made the decision to refrain from publicly distributing the interviews in their original form and in their entirety until we have converted them into formats appropriate for dissemination and wide-spread access to broad demographics, e.g., to villagers in Rwanda. Second, at this time we do not have permission to disclose two of the 49 original video interviews in their entirety. Specifically, one video interview has a phrase redacted and one video interview can only be publicly released at a pre-specified date in the future. The original videos of these interviews, as well as a redacted version of the former, are included in the corpus. Our architecture allows the verification of both the entire corpus (for those who have appropriate access), as well as the verification of individual components in the corpus; e.g., it is possible to verify the integrity of any of the non-restricted 47 video interviews even without access to the other two video interviews. Finally, we note that each interview is preserved in its original form and, as such, each original video interview is actually spread across multiple files.

**Standards.** All videos in the corpus are encoded in the Material Exchange Format (MXF), defined by the Society of Motion Picture and Television Engineers (SMPTE) in the SMPTE 377M [8] standard. The files use the OP-Atom operational pattern, defined by the standard SMPTE 390M [9]. The MXF files wrap DVCPRO HD video data using the SMPTE 383M [10] container standard. The DVCPRO HD video data is encoded using the SMPTE 370M [7] standard. This specification depends on other standards as well, such as FIPS PUB 180-2 [1] (included in the corpus as `fips180-2withchangenotice.pdf`) for the SHA family of hash functions and ISO 32000-1:2008 [4] for the PDF document format.

When possible, we have included electronic copies of these standards in PDF format in the corpus itself. Copyright law does, however, prohibit us from publicly re-distributing some of the standards included in the corpus. Because of their nature as standards, we expect for them to be independently available and verifiable in their own right in the short term; in the long term we will explore appropriate ways to include them directly in our corpus (Section 3.1).

**Software.** In the corpus, we have included a Perl script called `ictr.pl` to hash, sign, and verify the corpus and its extensions. We have also included an Ubuntu 8.10 Live CD (bootable) image that can be used to execute the Perl script. This CD contains OpenSSL 0.9.8g-10.1ubuntu2.1 and GnuPG 1.4.9-3ubuntu1, either of which can be used to produce and verify corpus hashes. OpenSSL can additionally used to verify corpus update signatures (see Section 3.1 for details).

**Detailed Corpus Description.** The document named `Corpus_Description-YYYY_MM_DD.pdf` (where YYYY specifies the year, MM specifies the month, and DD specifies the day), which is stored in the corpus itself, provides a detailed list of files in the corpus with sufficient detail to properly interpret the entire corpus.

The document named `Corpus_Description_Public-YYYY_MM_DD.pdf` is identical to the previous document except that it excludes the name of the individual corresponding to the video that cannot be released at this time.

**Vault.** Because the raw videos are not publicly available yet, we cannot directly use replication to assist in (1) non-destruction and (2) authenticity preservation. Our use of hash functions, to be described, addresses authenticity preservation. To address the risk of data loss or destruction, we

are currently storing digital copies of the corpus (on hard disks) in several secure locations, including one at the University of Washington Libraries Vault. In this vault, along with the hard disks, we are storing hard copy printouts of all the PDF documents contained within the corpus. We will also transition to more robust, yet still privacy-respecting, methods for replicating the contents of the corpus in the short term, before the entire contents of the corpus (minus the redacted portion) becomes publicly available.

## 2.2 Integrity Verification Architecture

Roughly, our integrity verification architecture involves computing a hash over the entire contents of the corpus and publishing that hash in a widely disseminated and archived format. Our specific architecture is reminiscent of ones proposed by Haber and Stornetta [3] and Haber and Kamat [2].

Our integrity preservation architecture currently uses the SHA-512 cryptographic hashing algorithm, defined in FIPS PUB 180-2. The hash of the data set is the output of the SHA-512 algorithm ran on the manifest contents (described below). The published hash format will consist of the hexadecimal-encoded hash value, with dashes separating each group of four characters. The hash will be published in high-availability and high-integrity locations. Section 3.2 describes how to update this hash in the event that the security of SHA-512 comes into question or a new hash function is standardized upon.

### 2.2.1 The Manifest

A key component of the hashing scheme is the manifest. It specifies the complete contents and order of the data set, and the SHA-512 hash of each file in the data set. By hashing each file, and then hashing the hashes of these files, our approach is in effect a highly structured hash tree [6]. The exact format is specified in this section.

**Format.** The Manifest is a flat, ASCII-encoded text file. The file lists all files in the data set and their SHA-512 hashes.

**Grammar.** The following grammar precisely defines the manifest syntax:

```
manifest := Version: version<LF>file-info*  
version := [0-9]+  
file-info := file-name:sha512-hash<LF>  
file-name := [A-Za-z0-9_/.: ]+  
sha512-hash := [0-9A-Fa-f]{128}
```

For clarification, <LF> is the ASCII line feed character. Brackets represent character classes. For example, [A-Za-z0-9\_/.: ]+ refers to any ASCII character A through Z, a through z, 0 through 9, an underscore, a forward slash, a period, a colon, or a space. An asterisk means zero or more instances of the preceding term, while a plus means one or more instances of the preceding term. Numbers inside curly braces indicate the number of times the preceding term is instantiated. For example, the *sha512-hash* is 128 characters, each one a character 0 through 9, A through F, or a through f. All other non-italic characters are literals. Italic characters are composed to create non-terminal symbols.

Note that since filenames can contain colons, only the last colon in a line separates the file name from a hash. Implementers of manifest verifiers must ensure that their parsers properly handle file names with colons in them.

**Semantics.** The *version* term is a natural number representing the version of the manifest. This specification only defines version 1. If needed, the version number shall be incremented by one when the data set is updated, as described below. A new version of this document must accompany a new manifest version. The only valid versions are described in updates to this document that meet the requirements of Section 3.1 (e.g., those that are part of signed updates).

The *file-info* term is instantiated for each file in the data set, including the specification files. The *file-info* terms are in the order of the corresponding files in the data set. In each *file-info* term, the *file-name* term is the name of the file. The *sha512-hash* term is the hexadecimal-encoded output of the SHA-512 algorithm ran on the file contents.

The manifest file is named `Corpus_Manifest-YYYY_MM_DD.txt`, where YYYY specifies the year, MM specifies the month, and DD specifies the day. For example, if the corpus is committed on January 26, 2009, then the manifest file would be named `Corpus_Manifest-2009_01_26.txt`. The manifest file is not included in the manifest.

### 2.2.2 Hash Computation and Verification

For robustness purposes, we have utilized multiple platforms, using a mix of OpenSSL and GPG<sup>1</sup> to produce and verify the hash. In particular, we used an Intel Mac Mini, a Dell Precision T3400n, and two Dell Latitude D800s running Ubuntu 8.10. For `Disk Set 1`, the hashes were computed with GnuPG; for `Disk Set 2`, the hashes were computed with OpenSSL. The manifests of both disk sets were compared to ensure the hashes were computed correctly.

The following commands are issued by the script to generate the SHA-512 output:

- `openssl dgst -sha512 <filename>`
- `gpg --print-md sha512 <filename>`

**Verification.** In order to verify the integrity of a file  $f$ , one must first obtain a copy of the manifest  $M$  and the published, authentic hash for that manifest  $h_M$ . Compute the SHA-512 hash of the file  $M$ , e.g., by running one of the commands mentioned above on file  $M$ ; call the string output by this command  $h'_M$ . If  $h_M \neq h'_M$ , then there is a verification failure. Otherwise, the manifest is verified and one can proceed by computing the SHA-512 hash of the file  $f$ , e.g., by running one of the commands mentioned above on file  $f$ ; call the string output by this command  $h'_f$ . Parse the manifest  $M$  to find the hash for file  $f$  contained within  $M$ ; call that value  $h_f$ . If  $h_f = h'_f$ , then the hash is verified and the file is part of the official corpus.

## 2.3 Hashes for Individual Videos

We have also prepared hashes of individual video interviews. Specifically, since the original video interviews of a single participant span multiple MXF files, we hash each of these MXF files using SHA-512 and then construct a manifest for each video interview. We hash each manifest with

---

<sup>1</sup>GNU Privacy Guard – an implementation of the OpenPGP standard as defined by RFC4880

SHA-1, as defined in FIPS 180-2. Several photos of video interviewees appear in *The New York Times* on January 27, 2009, and under each photo is the corresponding SHA-1 hash value; SHA-512 values were too long for this purpose. While *The New York Times* article will serve as an additional archival source for these SHA-1 hashes, and thus an additional route for verifying the integrity of these interviews, SHA-1 is showing signs of weakness [11]. Therefore we stress that a stronger degree of authenticity verification can be obtained by verifying the entire corpus with SHA-512 as defined in Section 2.2.

## 2.4 Specific Data Set Timeline, Commands, and Actions

For transparency, we describe here all the actions that were performed with the corpus:

1. Videos were captured with a Panasonic AG-HVX200A camera onto Panasonic P2 cards. The videos were then moved from the P2 cards onto one of eight hard drives in **Disk Set 1** and/or one of three hard drives in **Disk Set 2**, with the majority of files stored on **Disk Set 1**.

The following describes the details of **Disk Set 1** and **Disk Set 2**.

- **Disk Set 1** (8 Hard Drives):
  - Four (4) LaCie 500GB D2 External Hard Drives
  - Two (2) OWC Mercury Elite Pro 1TB External Hard Drive
  - One (1) Epro 1TB External Hard Drive
  - One (1) Western Digital 500GB External Hard Drive
- **Disk Set 2** (3 Hard Drives)
  - Two (2) OWC Mercury Elite Pro 2TB RAID External Hard Drives
  - One (1) OWC Mercury Elite Pro 1TB External Hard Drive

2. Documents were produced clearly describing the structure of the file system, and which files correspond to which interviews for both **Disk Set 1** and **Disk Set 2**. These documents were combined into the `Corpus_Description-YYYY_MM_DD.pdf` and `Corpus_Description_Public-YYYY_MM_DD.pdf` files.
3. All contents of **Disk Set 1** were copied to **Disk Set 2**. The group of files existing only on **Disk Set 2** (the contents of folders P2\_0048B, P2\_0074B, P2\_0075B, P2\_0076B, and P2\_0077B) were copied back to **Disk Set 1**. Note that this was an append-only operation — nothing was overwritten or deleted.
4. One phrase (which the interviewee did not want released) was redacted out of one video interview. The resulting video is stored on **Disk Set 1** and **Disk Set 2** and is stored in the MOV format. The MOV format is defined by ISO/IEC 14496:14-2003 [5]. This particular MOV file contains DVCPRO HD video, defined by SMPTE 370M [7].
5. Manifests for **Disk Set 1** and **Disk Set 2** are created, as described in section 2.2.1.
6. For both **Disk Set 1** and **Disk Set 2**, all files for which there are duplicates were checked for bitwise equivalence as approximated by the cryptographic security of SHA-512. No inconsistencies were found.
7. Signing keys (public and private) produced and stored, as described in Section 3.1.

8. Public keys copied to both `Disk Set 1` and `Disk Set 2`.
9. This document finalized and copied to `Disk Set 1` and `Disk Set 2`.
10. Manifests of `Disk Set 1` and `Disk Set 2` finalized and hashed.
11. Final hash of manifest computed and published in the public/legal notices section of either *The New York Times*, *The Washington Post*, or both, on March 31, 2009.
12. `Disk Set 1`, `Disk Set 2`, and printed hard copies of the manifest and all PDF documents stored in the University of Washington Special Collections vault.
13. Further dissemination and preservation of materials (ongoing).

### 3 Future Corpus Evolution

Over time, the corpus may evolve in the following two manners:

- **Corpus Extension.** Additional content, vetted by the corpus' custodians (see Section 3.1.2) may be added to the corpus.
- **Hash Update.** The verifiability hash (described in Section 2.2.2) along with the manifest (described in Section 2.2.1) may need to be regenerated in order to provide continued verifiability in the face of continuing advances in cryptanalysis.

#### 3.1 Corpus Extension

The corpus will periodically be extended with new videos, translations, formats, and other content. Corpus extensions revolve around the following two goals: (1) enabling the corpus to grow; and (2) realization that the way in which we perform future extensions might evolve over time. Thus, in this document we are only describing the way in which the *next* extension will be performed and expect the next extension to specify how one or more subsequent extensions will be performed. This philosophy allows us to remain sensitive to evolution in cryptographic techniques. At this time a key property of our extension methodology is to publish and distribute the cryptographic verification values for each extension as broadly and visibly as possible, and to rely on the ability of the public to verify the validity of these values. The first corpus extension will be completed by January 31, 2014.

##### 3.1.1 Procedure

The following procedure will be followed to generate the next corpus release:

1. All additions will be placed on a set of hard drives named `Disk Set New`. Also included in the new corpus additions on `Disk Set New` is the manifest from the original corpus (currently stored on `Disk Set 1` and `Disk Set 2`). The manifest must be named in accordance with Section 2.2.1.
2. `Disk Set New` will be copied to a back up disk set, called `Disk Set New - Backup`.

3. A manifest will be generated, as described in Section 2.2.1, on both the current version of the corpus and the extension. The difference from Section 2.2.1 is that if SHA-512 is showing signs of weakness or if a new hash function is widely accepted as being more secure, then a hash function other than SHA-512 may be used; the hash function used in this step must be widely accepted as being more secure than SHA-512.
4. A hash of the manifest will be produced as described in Section 2.2.2. The difference from Section 2.2.1 is that the hash function must be the same as the hash function used in the above step.
5. The manifest will be stored on both `Disk Set New` and `Disk Set New -- Backup`. The manifest must be named in accordance with Section 2.2.1.
6. To assure that new corpus releases are vetted by the original corpus team, digital signatures (described in Section 3.1.2) will be applied to the hash produced above.
7. The hash, along with the digital signatures, will be committed to in one or more high availability, high integrity locations.
8. `Disk Set New`, `Disk Set New -- Backup`, as well as paper versions of all PDF documents and the new manifest file, will be stored in the University of Washington Library Vault.

Note that all extensions only add content to the corpus; no content is altered or removed.

### 3.1.2 Digital Signatures

Three people — Batya Friedman (Professor, University of Washington Information School), Alan Borning (Professor, University of Washington Computer Science and Engineering), Tadayoshi Kohno (Assistant Professor, University of Washington Computer Science and Engineering) — each have individually generated a different private/public RSA key pair. The public RSA key of each team member is published in this document (see Appendix A). Each key holder is currently storing their respective private RSA key in a different physically secure location.

**Key Generation.** Each key holder separately generated a 16384-bit RSA key pair on his/her own machine, using OpenSSL on Linux or Mac OS X. The following commands were used:

```
openssl genrsa -aes256 16384 > priv.rsa
openssl rsa -in priv.rsa -pubout > pub.rsa
```

**Key Validity.** The public keys included in this document are valid *only* for signing the first corpus extension. The next extension will specify new keys and procedures for signing subsequent updates.

**Extension Signature.** To generate a new corpus extension signature, each team member will compute the hash of the new corpus (as described in Section 3.1.1), sign this hash with their private RSA keys, and publish the results.

The following commands or their equivalents will be used for respectively signing the hash with the OpenSSL generated keys:

```
openssl dgst -sha512 -sign priv.rsa <filename> > filename.sig
```



The latest stable release of OpenSSL will be used for this step. If in Section 3.1.1 a hash function other than SHA-512 was used, then that hash function may be used here as well provided that it is supported by OpenSSL.

The signature will be stored as a DER-encoded<sup>2</sup> ASN.1<sup>3</sup> structure containing the hash value and algorithm tag. Each signature will be embedded in a clearly-named file indicating the team member who signed it and the date of its signing. These files will be distributed with the corpus extension and hash.

**Signature Verification.** To verify a corpus extension, compute the hash of the extended corpus (defined by Section 3.1.1). Verify each published signature with the corresponding public key contained within the original corpus; this may be done using the commands below. The corpus extension should be considered officially vetted when at least 2 of the 3 valid signatures are present. See Section 4 for the reasoning behind this requirement.

The following command can be used to verify the signature, assuming the use of SHA-512:

```
openssl dgst -sha512 -signature <filename.sig> -verify <pub.rsa> <filename>
```

**Key Revocation.** In the event that any key holder's private key is known to be compromised, a corpus extension will be immediately issued, published, and signed by 2 other keys (as described above). This extension will invalidate all existing keys, and provide new keys and procedures for signing the next corpus extension.

## 3.2 Updating the Hash

Advances in cryptanalysis may reduce the strength of the integrity mechanisms presented in this document. Therefore, periodic updates to the hash using new algorithms may be expected. These updates will normally be published as extensions to the corpus (see Section 3.1). However, it may be desirable to update the hash after the original team has dissolved and no other officially-vetted corpus extensions can be made. In this case, any interested parties should be able to update the hash. All updates to the hash must be accompanied with instructions on how to go from the current hash at the time of the update to the new hash in a manner such that the public can verify that the new hash is correct.

The next or subsequent vetted versions of the corpus (as described in Section 3.1) may supersede the details described in this section. Prior to this section being superseded, if the team dissolves and there is a need to update the hash value due to advances in cryptanalysis, then the updated hash should be computed over the original corpus exactly as described in Section 2.2 but with one or more hash functions that are accepted to provide security greater than that of SHA-512.

## 4 Security Considerations

This section surveys the motivation and rationale behind several design decisions.

---

<sup>2</sup>Message transfer syntax specified by the International Telecommunication Union in X.690

<sup>3</sup>Abstract Syntax Notation One – ISO/IEC and Telecommunication Standardization Sector (ITU-T) standard notation described by X.680

## 4.1 Main Security Goals and Decisions

The main security related goals of this design include:

- i) *Protection against Revisionist History.* Future generations should not be able to claim, for example, that the International Criminal Tribunal for Rwanda never occurred, that the interviews from this corpus are fake, or that the interviews have been altered.
- ii) *Publicly Verifiable Integrity.* Since the corpus will be in the public domain, it is important that the public be able to verify whether a particular electronic copy of information is indeed part of the original corpus.
- iii) *Publicly Verifiable Authenticity.* The corpus is likely to evolve over time. It is important to signify whether a corpus extension is indeed authentic, i.e., whether it has been vetted by the corpus curators.
- iv) *Robustness against cryptanalytic techniques.* The system must continue to provide a high level of protection in the face of cryptanalytic advances.

Goals (i) and (ii) are met with the use of a cryptographic hash function and the publication of its output in a widely distributed, widely archived source (see Section 2.2). We use SHA-512 because, given the current state of cryptanalysis, it is unlikely that an attack on SHA-512 will be discovered within the next several years. If SHA-512 does show signs of weakness prior to the next corpus extension, we have outlined the procedure to rehash the corpus using a different algorithm (see Section 3.2). For robustness and correctness, the corpus has been hashed with two different implementations of SHA-512 (from OpenSSL and GPG).

Goal (iii) is assured by the use of digital signatures (see Section 3.1.2). We use large (16384-bit) RSA key pairs to protect against adversaries with large computing resources and unexpected cryptanalytic advances. Because this document carries the three valid public keys and its hash has been included in the manifest, the public keys have been committed to and cannot be altered. The requirement that any valid corpus extension be signed with two of the three private keys provides protection against a single key being lost, the compromise of a single key, or the coercion of a single team member. If any of the above conditions is known to occur, an immediate extension will be released, revoking the key in question (as described in Section 3.1.2).

Our system meets goal (iv) by only specifying the instructions for the next corpus extension, which will be able to specify the requirements for the subsequent extension based on any new advances in cryptanalysis.

An assumption made by our design relies on the “cry foul” defense. The public must verify all new hash updates and “cry foul” if an update can’t be verified. This is especially important after the original team dissolves, unless future custodians are chosen, since no one will be able to sign updates.

## 4.2 Perceptions of Credibility

Some of our decisions were made not because they enhance the credibility of the corpus per se, but because they enhance the *perceptions* of credibility of the corpus.

**Decision 1.** Disk Set 2 contains a verified copy of the corpus, but in the original file structure. As such, it is unlikely to be accessed. Additionally, the verification scheme is not designed to handle

it, although it could be verified on a file-by-file basis. The decision was made to retain Disk Set 2 despite these limitations because its absence might be perceived in the future as undermining the credibility of the corpus.

**Decision 2.** An “amend” only policy was taken toward material on the Disk Sets. That is, if a Disk Set contained duplicate copies of a file, the duplicates were preserved rather than deleted. Thus, we can make the statement that once downloaded onto the hard drive no material was intentionally removed from the original Disk Sets 1 or 2.

Finally, note that a “preliminary-final” hash value for the corpus appeared by accident in a public presentation regarding this corpus on January 27, 2009. This “preliminary-final” hash value was derived using an older manifest and an older version of this document; we have included both the older manifest and the older version of this document in the corpus so that people with knowledge of this “preliminary-final” hash can verify that the new hash has not changed any of the actual interviews. This older version of this document had a different public key for Batya Friedman; it was changed in this version due to her forgetting the passphrase for her old corresponding private key. Any authentication **must not** use this public key.

### 4.3 Cultural Norms and Conventions

It is well known that as language evolves over time, certain words, terms, and naming conventions change in meaning and correctness. This section serves as a reminder of the context in which this version of the corpus was created.

**This Document.** This document is written in American English circa 2009.

**Interviews.** The interviews were recorded in the language preferred by the interviewee. The interviewers spoke in English. Interpreters were employed at the request of the interviewee. Of the 49 interviews, a small number were conducted in French with a French speaking interpreter; in one case, an interviewee spoke a small amount in Kinyarwanda.

A good deal of meaning, subtly, and intention is conveyed through gesture. Thus, the cinematographers attempted to capture both close-ups for facial expression as well as hand gestures and other body language as appropriate.

## 5 Acknowledgments

This work is based upon material supported, in part, by the US National Science Foundation under Grant no. IIS-0849270, a gift from the University of Washington Foundation, and an Alfred P. Sloan Research Fellowship. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation, University of Washington Foundation, or the Alfred P. Sloan Foundation. We thank the University of Washington Libraries for providing vault space for the tribunal materials.

## References

- [1] Secure Hash Standard. National Institute of Standards and Technology, NIST FIPS PUB 180-2, U.S. Department of Commerce, 2002.

- [2] S. Haber and P. Kamat. A content integrity service for long-term digital archives. In *IS&T Archiving Conference*, 2006.
- [3] S. Haber and W. S. Stornetta. How to time-stamp a digital document. In *Advances in Cryptology – CRYPTO’90*, 1991.
- [4] International Organization for Standardization. Document management – Portable document format – PDF 1.7.
- [5] International Organization for Standardization. Information technology – Coding of audio-visual objects – Part 14: MP4 file format.
- [6] R. C. Merkle. Protocols for public key cryptosystems. In *IEEE Symposium on Security and Privacy*, 1980.
- [7] Society Of Motion Picture and Television Engineers. Data Structure for DV-Based Audio, Data and Compressed Video at 100 Mb/s 1080/60i, 1080/50i, 720/60p, 720/50p.
- [8] Society Of Motion Picture and Television Engineers. Material Exchange Format (MXF) File Format Specification.
- [9] Society Of Motion Picture and Television Engineers. Material Exchange Format (MXF) Specialized Operational Pattern Atom (Simplified Representation of a Single Item).
- [10] Society Of Motion Picture and Television Engineers. MXF Mapping of DV-DIF to Generic Container.
- [11] X. Wang, Y. L. Yin, and H. Yu. Finding collisions in the full SHA-1. In *Advances in Cryptology – CRYPTO 2005*, 2005.

## APPENDIX

### A Public Keys

The following are the public keys, which should be used to verify the authenticity of the next update. See Section 3.1.2 for a description of how digital signatures are used.

TADAYOSHI KOHNO'S PUBLIC RSA KEY

-----BEGIN PUBLIC KEY-----

MIIIIjANBgkqhkiG9w0BAQEFAAOCCA8AMIIICgKCCAE4EnwqY3oEpAwsN5PmbLX  
cLwRfLORj7MsM3f2j0Tp+CTjFYspczBtJB8hBojr45nN+NN7ZGKXwIaBjEsxcctw  
MudcCrXmg5X54CzbBoTixGi+GUZF1fm+ZZOCofnS2pLM1A2mgE8KX21kNyU8kmtx  
9FUPqYnh77Wz9pR7m9tnbzafZGGxF1udIt4HoMvJQyAxy0I1Ek1zeVKKD4kuwvH  
JqZbXU6CWWDOkTxiY713VFr4+ZDdVSD3mRBm/fGwAPg8k2pfWZ3w11F2mJjgA4R  
9Y0anxow7GdiEs5vzbXVwScE80/kD9WjP4KK7+RXWAWPLChPOXWFT+R5JHE02DCW  
1dWtduhBgoKKhEX2LBjuyS0Bq8XARuD1PONncIqKk3ufGPG8kGs0gL8NNse8qhG1  
YNgIbR16ffGx9SoI8m6GkGVXbhrki9XWQzjJUIwIw5mbMX3YBEetP/C76IEfPdy/  
rYDymDJ9GkaWGbODWhdyx4JYrSdxVP+7Ddid0QjgmpWgJrT1Eo6gZ+mygNiVkDMU  
gJcK8VfeyNbIhafAGb2hB9xuzC2DIhs/7qJmEax1oq6RDmEDIjC7uQIw3mth7Ked  
+kTCat5bBsUgK6UACmYYgdYc0AxmIF0dUf10T5zKzqpZV9THE89e9+KvkOuOf4UW  
/EBiGrapjC2C3YKix4V1+V78Z8GPF/7PQiibs/FKrE8c99yF++FfDI7kBCyDtr4  
gJK8XN0QScmpqF0Lf7243iLaLd9WzugdronzMGDU7DhLls2fFymAp0a8yzMNma53  
Wm+uzw0oIGTzuIpn3Dv77Y0BiXaxHLyxcQ/KNjtxrFQEaMNGDE6pDCDdr5ldPKNa  
1Qm7z0cDV84ugivm94hZSSUx54cDbBsmHYkWTz+pNcrdddqoU849I5pHK8xHm+sq  
Gvjv74r99qKLGvzvW+Z79uvFFeQvmTQv8ADcfzrgFPkzv5NLihZMe2C/+IlwnJjH  
H20s2pXjR1KgzpCFhprK36sTu3pxi6ZkaWc3U3LRAOpw2T1XKuM9IqA7PTdMct/D  
q1DD+E+DgCIM5b+74rLy+GFiybstbDsUz5HI6t2TiYT/Wz9ZVo8iUXfv0iWUlrDq  
Uw4h6Z6+82F0Kvpgw4WyQVZ9mCIaJ9CEYvCXUjEX9H6o3w9ty0/feQ/n9o7mzSK3  
PY1Dm8CiHV01ZPTSbprrgE6MZsuj5Q41NiDLd5QRNX8RscDqtU19taDK7XoXHWqZ  
eMTt10E4ISGd86IXHk1063yus0qd57bEgatMyq6VdaP/5paIk5N7E4oZSYZDP2Eq  
IOWE8+ho+3/FdmjP+3vLP3tNOMNcf2/CwlJWUVkPphyTzQJmB3ST6X47+lhIfLD3  
xiZVtn/YwyONdxL/DcLD0zbOnsMLTNN53TkM2/NYkVp110K8GrIRD1jDnRhtHoTa  
M3fTjMH7NPIUf2J9Gnt+bUe35TQI+k8flp5cD5QUJrdWBxtjujma30xk/92LFRxe  
05fHhdB14+WjzQKX0LjJpLKDbst/TpZm6Dd8Q4KktgCRjFmLGUQ0j9NSfv3mA28E  
zw8hsoLzFuskfMkY2Efms9DiYSfd5kMttNjYU0kL319Wq9eGEN12Hc8VPPNF9V0c  
E6Ym8ziARdk6MBUrNDfmUmN6yfJOP98D0gr5QZBgZFx2N7LD0hQfJr4W1V1Z9CZ+  
5rVcS8cH55BqdQnrV8R7INr1/ito1Taf/YG4QLuYHpzumnu00t8sIuFo76roTfw6  
QoDoMWRx5FqkNIQAEEV6pdpj7YhWMXeZ1Eli/j+7gq0EI6Z5/mirgTzwqUn7X3PB  
duIl3idugf8/eJtm3wumxotVW+tykOYGBE06abHOr7Y5xfGEWLZyL1Db18RMefGm  
xvoMLj2IctrHeIrrXV1g5+fM4LHZ5/EvdDbMOYackssmiuUuCOTxPmQ4zK+F0fZS  
bT1BsVQRckFoeWYF6BnPFJVTqIfyH4BHB6YBdo4k5vxGAZC8TR3z92cx7MXACTs6  
YCYEFFFR13502xQ9uuRjipXnfdc+Nj0BiCVSSj8zlnB/J/6mrh9IFeaCL2UN54z2  
rzMsVuFuFQe6gxr2t0R7R/CHY3guwwW/kdi7gVOU3C8FKiW8ZSAiP6rnAWGZz+R3  
+jEKC05KrK9kBQldxPOEJN3iDMchmyxPT4i1Jybi3l/7Jj0tIUYc+flxyX6P+0c+  
xPS5zgiLwRxQBdL5/uhN0SnTY5oVyMJfUu41PQ2j774ooCPKto0cQvFkDRH+d7BA  
e4mDow9K472Iie0dqjnvJmRQMr1cW/BhLV6eId7jQML5vqWXY7wCquaWxu3RysCs  
ajBlKuGfB0cHCBB6mg0J8KbHrzLc0sTT313KxB5VEbm5M9acrE9q0s6bBivwzNXu  
z6wDHTLIqRYxI9bI5xrR8+Emsi9ld8jRoF1gbsVzEyG1hFDdBqPRvvneyx44iaR  
u0LY5sZ8TydXVzmX1kaYM62qXEx8GRwza1fQiLtiUdHJhFarziK2eXuDipcL5cZ  
ii+0QiuPJvGkP1yuc0tqpYS6M1UItXDP1cf6G94jVIE+mPUjSaTkNm17FN1jDqAb  
GXoqA9CLPwuWENVGTSPZuQmPufS2e2iN7Ra0x+LQuxAQ12YVzfUGinAg3RuHxesn  
CEj2+FVygZrsY33iYqWcyykd3p1/gIFsaUa4if0sf7nhs70tNSktFztzxsMrdgTX  
BHxgyNRrWMDsDi8SVRRa49ECAwEAAQ==

-----END PUBLIC KEY-----

BATYA FRIEDMAN'S PUBLIC RSA KEY

-----BEGIN PUBLIC KEY-----

MIIIIjANBqkqhkiG9w0BAQEFAAOCCA8AMIIICgKCCAEAE8V2LK6pB5qNqETHyfpjj  
FBICLKAqX4kT1/2MbStx+4NSKDt7XGVBhYepkV59srBMyVovZzJX49SquduzUZ2KW  
yv9yC2XJy1Fu+qImPc70M+I1aSd9broGA1jzRJsZ/zrQqDdbpRMnIIPsgzoZdFfS  
/oAUePE5/iiKel1gim+nvCU2/xM4IJMeZgX+5hN41DiRis/PAbMGzpcGsC8enw6  
R2bB3AsuVzoYmJQRPUbKvXTy3ht8p7F7xWl7Q8ytW487xmxTV0v2HdKK6Yuc0YtZ  
MpqkYHFHbrWnsF5RNz1E7dVTCNI4tSK0VV2Vfy1epVePgrNSJOCmey2CL/sxXsUE  
EjWYK5an+NtAp2SZln0wyN/BxUNlsqBng8TVq/mG5CBmhmY2YqBTuoLQzr2bebH  
uv5A54ShU8N7+Xk2iIN1k6bD/K63zvTfyvzxT1Zw/epjUEuvW8JfnSMnOU2xC4Dm  
M18IfXda981WyWLDAPqEW2mjgEIMF2ydhW5WnewfDMkFE02vuYf/bxL+R1VBzcy  
DY7EBgQmw/81t9naCE1TlIG/agk0LC45Bm9eNQF7CRpNwbxGFI8aTDW+jju157yU  
LAm8osA187IQYXGyWCFtGv1iBS/wzlaWPcvbp9FF5tDG9BJvxLAWVPer6hJLx0Nx  
MG6fg5zRdP//1EkMG8FKyaabteLgd5fbfTS965hHnpGZAgTMIjzdeFezpl/FfE+E  
D/kNEDXmIJb5bg4AUjGt+mqz9WIKRnS8Cm0Tmw2e2rQbBC0kxi20V51RpIevjR8I  
WH/MKRvLR/YEnleIcel/DVXs9GMS5m+h9Q7IBYEvqtRKVP00bE5PMNoOmLAOnfBt  
Wy/5q4klqh09ZSCvpVC1+UFA24eMXDVe1GaXbDhBpZibEY9Kqntx84cgIZpAhjK7  
7caR6wRLH0t42td2IuIizj0X1+3JeyH/j9fCefuslV1WB/oUy2je5P23aISbah2T  
KTQjPjUmOYchpfKp+DLZf/LOE84InDR72s4+DODWEpoJb5yL3Z9xIH9z2P+GcDcn  
Lgxuq/i/rcZdHzJCaRT53u1TieWEK/zG/L17RpkV1ToKxfMhc2dgp26RmDLmIcex  
p33ihCMLHzv4eo6/or8aM1RC28M0h0jqT1rLE6pGjs57xfe9CQLW/eoKAQI6mvib  
GyYUQe9EXvAjK3v2nZ4uhZTDVesY1re3/x0Y1Yot1VbNRTGYOWHYGN+S2izwDkiY  
rwyiHGK1gc0IwbsGQV6Ndiex4ulVh9wXQp0YPOdK1YHu0D5ulz6jX38YAthTwdyy  
dUpRkNAD9S0JUMVmGYmi50GA+h1vORKuXUYoIVVMBEhdJ9Tmrh8fJ0sJlnbt0cqR  
qZ8HZbZS8wHH3n/ThGyZ7zclXV37sXSHN0cJ4TMBIL53WKKXng2BzFGgjJvQRM1F5  
Hw2AjSdHSZOimGpvtb/OzlmH49eNlMtnQ/yeDyMzCGE+PPUNLiB4n5jICWfE+MUD  
8KHDl0sVhiDrtcwUvNkmd8RD6YtSlls2+gPwwfDq3xhvMTtfhfMR4+GXuMVSjggh  
HvnCLHokWG1500HETSaxW0GMhqqvGMOLK6vZ290vIed09Kk05WZEMWSh575j5985  
CqRLc14tHxgZA21bDifh3JdUq+NUKClkM6sXkF08SPDPRxNsBsTQJpKz73PuB8aF  
hkAqRxs7dtYhcyLEvP3yKX5W0wqiWkedW8BPWChjEmMkbs6crTPXkkNbXM3Gu31  
p8XSTzrM+gTqggCn9VD5rmsdtZ+fa6isLqjLf0gpJgX9nlegcIPkZ9GMdTP5zF1N  
oCP5s7H1lw5P09pN8DhXY6g/2iH0i193hKsE/uDDc+eNnW5Q038HwevRItTW5H6  
tr1gbAS9X+ESVPZHmHK1Za0QZiYvZnZ2sGMDs1NhN8Uer8M474gVIt6tMGr3AGo  
ozDdHXstD9QUMUQHwSx7i4C1RH3tqY5A6WgHQzzfubP4caaFNPMY7B2w4sXQfXa  
zicwJsTAQsF7PwmQMZ8rayLUY9omR/u00W2MFROMXKRYvsm3HD+i9rw6ykmgVOMx  
n9LKzB0oLy0g7sh9WTQc/vH2qAUZvud0s9xh35m611Se2gp6CgtA6EHLUderhEx  
h182NDcx7GHSdrbvUQR5B6jNyB7mUWCs0TEhQA3RUFjHU09uyrMW8iHFfXF6FKK  
Cgv0uox9C0mmJscFFU0nKpg8hE34v//Qap5DoF4wGUTqa7KeXd8vHhaHFN/jZd5A  
gk5PEe7dQ1bsb5xLrIxV4oNtcwudqFYps+k45JSNJ3g6sHfLokUYGLvGh90LA7Q  
JuKWB8dApfqVqUPYIcPlI69k9U8NxSa/1uyeEEBsDCvhrMSbfGPxZKz2ZEdG1ZzD  
CnA2kGi6w6Cjizypm0l5wexX8CAzQvsKRqoubgikhcZs0Vmg+hzXv+tsyNsbGCZ  
z1cMSJ259G5vSfmZDVYRH0oUb6Qho/oZ6D8KvYjRp/QVJuc1RCC4BIkudW3AuCwY  
7eSTN2TiAg7sXESCft7S38iXACMKI2x/f6MZ1cndY2ZP9YTE7U4nnDJ6qNsxYmY  
3vUzy9ok3Yy87bYqUO/GBxeCy/F39o2qu/PugX78JD0mks55hg5GqeKiEzfDpSFT  
OGE9jmfhwfjNxzTExmFnGaiYIgi3zYLRJHBqAy17aoFBNXSjs8sWQ9b8K/cCn8h  
zoLEI9Phq68Ly8zitkdw3TOCAwEAAQ==

-----END PUBLIC KEY-----

ALAN BORNING'S PUBLIC RSA KEY

-----BEGIN PUBLIC KEY-----

MIIIIjANBgkqhkiG9w0BAQEFAAOCCA8AMIIICgKCCAEAw37R90t0M+hrkV5wyJuG  
uZVvHzaBlqC62NAkCT0+7y96l7woA65vkJy9wwiQhxFtstcH6vsh9V20BcbnFBcv  
zlebfDG4dfB5Z7otfqy6/s+QBxNA/jNbV8yAcB2l279PIjsk2Dxs1XwfEucwmY9X  
nBtSFyIra+5091bRp76G9vccxt0ZyBuEV3ZPPdiTfJnEhxMK27XtMt0QNM6Banu  
D0e7mWBe0j69ES2tCdS3hhj0U4+nkXw0crVUyPV0noUwGEsJUsNs6ChBsw82Gqdd  
dcny7Q1kw1VFa8JpBHwxxcW1GDfNh2kYiS0Zl1JxD1NbEaA3K7N03PCIE3ySUMbv  
fZadqy4/jErBV2Yx4CGStP1PM2yk/wzYd4S39M3xONL34pcI3vt/e80sCD1DUUYi  
boIi+ypAyQomi47ncNnGgURp6NyWt82GSr0srxU5pNh31PR+BoNvAJGOY7+hVXgG  
vwqzdpixQmzIKhgTVUN3GLCNPkrzQNWeLWtDiV36pFSV3a2YaKf4fqyF28GbaH2d  
We1HIZyMWNr8LfiK0Kv34GrPmBgaQXq9/UiuJndRbeMELteDjyYmSuW+GDIK8AP  
WOS9r7HBmb2o2tfoBVx0PByqxHp10Xt2N4zemndqfX248hGdZVthloV6SAyT1JxD  
gjAxWTJSf0pK5DgBuLzG04lwtV/hBrUtR8Pb54Y/FSIHsfaTFf5q66M4X6FzCpU7  
Gk8zYNLhjd1GuAw6ij1dRzrDuPVQ/dAnc00HNOHFxwpYZBL0KWuL78JNzbZx6b6D  
J/OaUtJx/RyssWE2FB+1vKVvmrRLopmzrIG6b2Ksd+vkF/2TZ7vavsQa3wdx17F  
hY49JfSFcMP0gBbusVUWwEMP+E41bIsZfy5fq0oS//mFafL2ocBYw/crm7o0tFc0  
7pIow0U1oa2M+ICNrLxbH+3ux1CbDSrYEDUw6I5nRKHZVcfWvom0WnfWNyAyQJRM  
T6BY/7tXh2l9WYZ75uM2FsLaIL91uWS73b1bLrr3m2jMzPQ5FreMvz2ix7IVVsAb  
XhvGFzfTqstUWLe9ZvNH2JCvsxX8WWeq5NTYnT1G1JStcvi/VvjHUsphjyCeIT8l  
wgERxNrOu45DHczSOTqcF+mk0K8m4xmtAG00bjFyk9r9WJNg/cOTSx6HImyfybjy  
axrg+ZHeFdfgvlhpg4GzfxpK6mVdJJalQfFy2v3fz9D0iPQM3Pvp0PBCKkWKCPiW  
PZsPRCqkzyrF0WWGtHockkWjLG5u/lzfbZjHQLlupd6oJuKx79U0w8TfFtCkfnx  
MGntQRwBmNRhLOGLNfrJQrTsSFAczJU26JNCA5Kw6c8i8L8qDG5P5/hIR7yQVFOG  
hG+ny5yxLMSTFhkWgkdQGuonH5ZiZp9Vb+uzS02XwJXvJZvvYHHOGFmOhTl4ESYH  
zLGW5mdAXYSetcaHwIk5NG9C0RIVpSplCX5D21s/74x9mIzlHjBegq2tcaAzyWkR  
Cz72ZVHwu0MOL8K+Kb6ZAPy4F6xsD6bjHfj1u1YIKyQ9d7ToqXsDjXXbppK1t14w  
SkfXeN/Pzpl7qWDreu6pUWQtBWqgm/spyIkQJMFN6BRJqM0JdYAMB1z1DY/iLoun  
5PBmzpIP+5RJa3ZCaUiBAkLiQpNnJA03oG1q9jn7EpuKTiu084sh4A+QoJ7smyxX  
Cj2QmPRK3ScJTFBLgSB7AD/V6gMG67bAqGIZvjQwVxWTUXR9hwoR+qnubyT7v9I  
8Qz05bFJuXNiEOgEEI1emHhF9ABF+03JbpiS6bBfoqnwqtVg0kPjr3QYxo6NLse3  
aaev9c74ZDNAPWRziK8+vGCxGI7FAGFIIf2ode6dna1Mtc3+PyeJzMcJFu5wZ0uuh  
yydk6e/OaEPS9Aen4uly68eI5r11cPWk/zm53gVd6cFeAlGSmCv4TD/mBk6b5KTz  
xjI2AKvqHwsiiylxCMX7Nha97fbUA0ZtUd+8lhMNCiKK95IF2t4KIL+i0Te2Mii  
x0zrcyl6yVpOynna3NBSJJKCNmIFpgBgb4dMwXlM34W1iioDt1j1LQPseVS1yDwt  
ASYrlzhMIe6NH0sQXoq0T/GYDgVQCUAvpnBdiuoYTtvc9hPYDpavMhXR+EDr1fdd  
xGORTjhv4rhGv7bUJ60elsPam4zhZjNi5AYp9fx/VokkKqzhqwhTH4PsZmY0pYgy  
02q0V1dn7XTrwVtZsA8VzeRQ0nu5oKR/7rTDihp2ungocks1h7VogMyraca01F3R  
WNZLiNX7dGwPaLVHgluNh16/+MfwBM+9+Y02GYbbv5n0bPG7B9cWy/9ITXHLmK0N  
bbrJQW4Bljw2X3ETtX5VCJRzWViswzHpGef4VBtp98zN7uf530fIMOEPFuYs40cp  
iIzFo04wPJRWpfZmJCWxmYIac7tJNXdU6Gh3a05c1K0oTN+Of/ikaKLCs06J+jg0  
ri7cmt08DIVcLkMeGVAjuZnN/hN2+lrU6PgfV46DwnbzyHf+RtKtEaibp167gtL  
odBVgWzbiH//1zUchD7Y8VRcGMD+ey7C3K0rYrukX99Ixpfi3ZEM40GK5b6X9x0M  
xf8U018zhMAQUdDvj0Th7+17ilUexL1XiSGOQ5ApQbV6zoNXZEyz/MDZDb+kgTuv  
Y0m87p+GeaaD5tCgY/HfuxKfnTzA4w9rtF3AJkCaVn1/rMVjHmJ6uKq+dpCPM+47  
1bLs0nfZfkcv1H65FkU6z4MCAwEAAQ==

-----END PUBLIC KEY-----