January 27, 2014
PARC

Reconstructable Content Objects

Marc Mosko[1]*

Abstract
In a typical CCN deployment, a system encodes user data in to Content Objects. Each network transmissible content object contains a small piece of the original data, say up to 8 KB. After a receiver as all the constituent objects, it may reconstruct the original user data. If the receiver wishes to re-transmit the content objects, it must save them, along with the user data, because the content objects are cryptographically signed by the original publisher and the receiver cannot reconstitute that signature if it throws away the objects. This leads to a potentially large set of duplicate data on a user's system to store the content object representation and the original user data representation. We describe a method to avoid the duplicate data using Reconstructable Content Objects.

Keywords
Content Centric Networks – Named Data Networks

[1] *Palo Alto Research Center*
*__Corresponding author__: marc.mosko@parc.com

Contents

Introduction

Reconstructable Content Objects store the original user data as a normal file on a user's system. This is the normal form an application would need to use the data, such as a JPEG or a game's data files. In addition to the user data, the system stores a set of metadata. The metadata describes how to publish the user data, such as the number of bytes per content object, the timestamps to use, how to name the content objects, and other things that go in each Content Object. Finally, the system stores the original publisher's signature(s) for the data. In a system that uses directly signed content objects, there are many signatures. For a system that uses secure catalogs, there may only be one or a small number of signatures.

1. Reconstructable Content Objects

Reconstructable Content Objects consist of several components. There is the underlying user data which is chunked in to one or more Content Objects. The chunking is done via a set of rules embodied in a metadata ruleset. This metadata provides all information for all content objects except for the cryptographic signatures. Finally, there is a set of cryptographic signatures, one for each content object. This is a necessary and sufficient set of objects to construct content objects from only the underlying user data.

Fig. 1 illustrates the process of creating and then reconstructing content objects. An initial device, noted as "Device #1" in the figure, has an original data file. Using a metadata ruleset, it constructs and initial set of content objects. The metadata ruleset specifies how a device fills in all the fields of a content object, such as the Creation Time or when to use an End Segment field, and the format of the CCNx names. The initial device then cryptographically signs each content object. This creates the set of signatures shown in the figure, which are actually part of each content object.

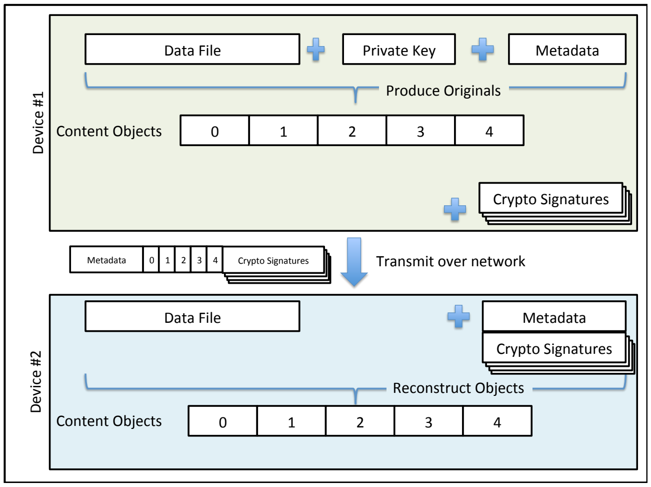

Figure 1. Reconstructing individual content objects

The initial device in Fig. 1 then transfers the metadata ruleset plus the content objects over the network to a second device, noted as "Device #2". The second device reconstructs the original user data file and saves the metadata ruleset plus the cryptographic signatures. This this small additional data, it may reconstruct the original content objects by following the metadata ruleset and plugging in the saved cryptographic signatures.

In a variation of this system, the first device may use

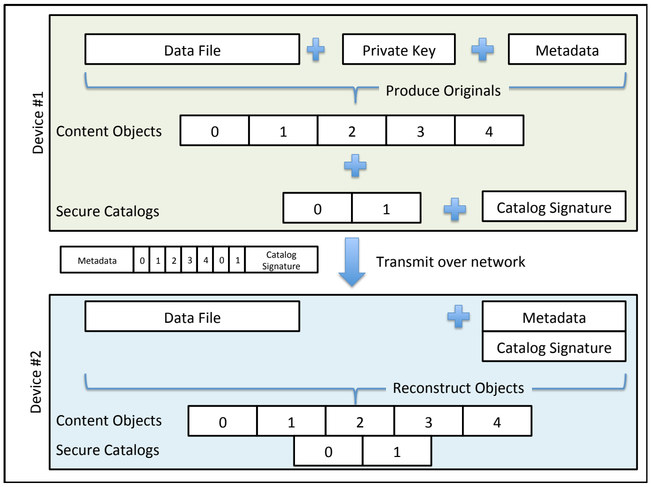

Figure 2. Reconstructing using secure catalog

a secure catalog (sometimes called an Aggregate Signing Object) to authenticate the content objects. This scheme, show in Fig. 2, only has a signature on the secure catalog, not on each individual content object. Therefore, the state transferred over the network and saved at the second device is potentially much smaller than the system where each content object is individually signed. In this variation, the rules for creating the secure catalog are in the metadata, so the secure catalog itself does not need to be stores.

2. Conclusion

We describe Reconstructable Content Objects, where a device stores the original user data file plus a metadata ruleset plus cryptographic signatures. Using the ruleset and signatures, it may reconstruct any or all of the original CCNx Content Objects to transfer the user datafile without having to store redundant content objects. A variation of the system uses secure catalogs, which significantly reduces the state stored in cryptographic signatures.