Collecting Metadata on a Instant Messaging Server

Alexandre Pujol, Christina Thorpe, Liam Murphy

University College Dublin
Performance Engineering Laboratory
School of Computer Science and Informatic
Belfield, Dublin 4, Ireland

Introduction

We want to show the importance of metadata leaks on a running server. We are working on Instant Messaging (IM) server use case.
From an attacker spying on the server:

- Intercept the metadata and retrieve sensitive knowledge.
- Also works if all the data is encrypted.
- Metadata can reveal more knowledge than data.

Motivation

- Enterprise legally obligated to protect the privacy of the users' data.
- Breaches of privacy can result in significant fines, legal action, and damage to reputation.
- Eg: Medical Records: If an oncologist access you records, you might have a cancer.

Our goals

- Show and quantify the leak of metadata.
- Define two types of metadata.
- Propose a proof of concept **attack** to retrieve leaked metadata on a server using live forensic method.
- Propose the use of an Oblivious RAM model as a solution.
- Provide a test-bed in order to test the security of future construction that will prevent the leak of metadata.

Related Work

[1] Privacy issues on smartphones because metadata are not protected.
[2] Extract information from encrypted YouTube video streams.

Security Model

We consider a simple Instant Messaging (IM) application. The IM server runs an IM instance. The messages sent between users can be either encrypted or unencrypted.

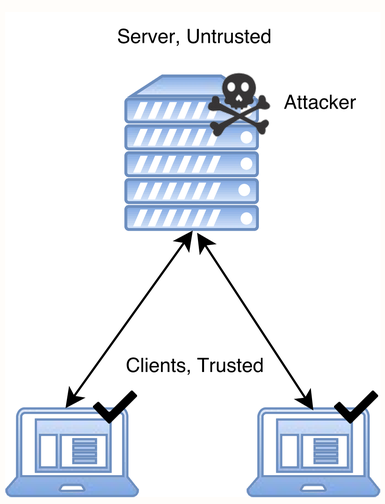

Figure 1: Attacker model.

We consider the following attacker model:

- The clients are trusted.
- The server is vulnerable to an attacker.
- The attacker has a root access to the server.
- The attacker is looking to retrieve as much metadata as possible.
- The attacker is a passive adversary.

We are the attacker

Metadata Definition

Tangible Metadata (TMD)

The metadata that can be written in the server they can be encrypted or not. They might be needed at some point to access the data.

Examples:

- File name, path
- File physical address,
- Time-stamp,
- Owners IDs.

Intangible Metadata (IMD)

The metadata generated by an action. We do not find them in a written form at any moment on the server drive.

Example: When the server is uploading a file to a client, the packets sent by the server reveals an action is being executed. This leaks intangible knowledge on the file owner:

- The time of the action,
- The type of the action (read/write),
- The action owner,
- The pattern of the actions.

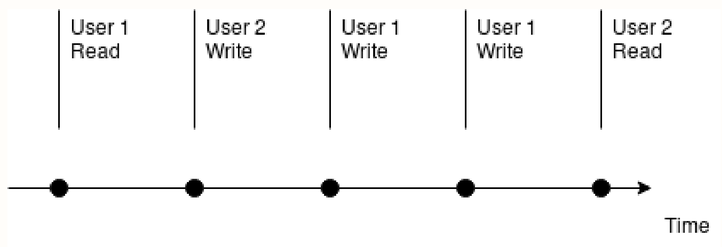

Figure 2: Pattern of intangible metadata leaked.

Proposed Attack

How to collect metadata?

- Collect the information on the server hard drive:
  Easy countermeasure.
- Collect the trace leaked by an action on the server RAM:
  Complex but hard to prevent.

Experiment

To analyse the state of the RAM of a running server in order to collect metadata. Our method uses the Rekall [3] live forensic framework. It works as following:

- The server runs Prosody, a XMPP server,
- Two dummy users simulate a conversation,
- The attacker run the Rekall framework to analyse the system,

Results

Although our detection system is in an early development stage, we are able to retrieve the metadata leaks:

- **TMD:** An action has been executed on the server from a given user to another and over time, the access patterns are revealed.
- **TMD:** IDs of the users.

How can metadata leaks be prevented?

Future Work

Oblivious RAM

Oblivious Random Access Machine (ORAM) [4] is a cryptographic construction that allows clients to access encrypted data residing on an untrusted storage server, while completely hiding the access patterns to storage.

Goals:

- Show ORAM techniques can be a good solution to metadata leak,
- Use our technique to test future ORAM scheme.

Improve our attack

- Test other kind IM server,
- Develop a new metric and method to quantify the leak of metadata,
- Test this attack on a ORAM system.

References

[1] R. Dubin et al. I know what you Saw last minute the chrome browser case. 2017
[2] J. Mayer et al. Evaluating the privacy properties of telephone metadata. *Proceedings of the National Academy of Sciences*, 2016.
[3] `http://www.rekall-forensic.com/`
[4] S. Devadas et al. Onion ORAM: A Constant Bandwidth Blowup Oblivious RAM, *In Theory of Cryptography Conference*, 2016