Securing Cooperative Backup for Mobile Systems

Nouha Oualha & Yves Roudier
{Nouha.Oualha,Yves.Roudier}@eurecom.fr
Institut Eurécom, Sophia-Antipolis, France

Abstract – **The use of mobile devices expands to various services, notably multimedia applications, resulting in huge quantity of data generated. The traditional backup function is relying entirely on the sporadic connection of the mobile device to an infrastructure. This is not sufficient to backup all the critical data handled by the mobile device. We propose a new alternative for the backup that relies on the cooperation of nearby mobile devices. However, such alternative raises, in its turn, new challenging security issues. We present these issues and how to address theses.**

I. INTRODUCTION

Wireless mobile devices exhibit increasingly powerful multimedia capabilities and built-in applications, resulting in large quantities of data being generated in the field and the critical need to back them up. Usually, most of this backup process is carried out only when mobile devices are connected back to an infrastructure. Given the fact that these devices are particularly prone to energy depletion, physical damage, loss, or theft, there is an obvious need for a new alternative to protect these data closer to their production time. We suggest that the very proliferation of mobile devices represents the perfect opportunity for finding nearby mobile devices that may cooperatively participate to such a backup process. Such a cooperative approach to backup however raises challenging security issues.

II. OPEN ISSUES

Cooperative backup relies on the interaction with unknown nodes, hence under no prior trust relationships. The interaction is based on a piece of the data, to be backed up, handed over to autonomous mobile nodes. Nodes should participate into the system cooperatively and fairly, however, nodes may misbehave in various ways, with new forms of selfishness and maliciousness.

A.Selfishness

Selfish nodes do not cooperate in order to optimize their resource usage. There are two types of selfishness: passive and active. Passive selfishness corresponds to nodes which do not participate to the backup service because they are not motivated to use the backup service. Active selfishness, also known as free-riding, corresponds to nodes that use the backup service without contributing their fair share: this may for instance require the collusion of several nodes to prevent the depletion of their storage space by using one backup instead of many as intended by the backup mechanism.

B.Maliciousness

Maliciousness is an active attack which may target either data or the infrastructure. Malicious nodes aim at destroying data backups: this may be done for instance between a set of colluding nodes, that associate to reduce the number of replicas of some critical data for instance, thereby reducing its potential availability. Maliciousness may also aim at destroying the data backup infrastructure with DoS attacks (e.g., flooding).

These threats to the backup system raise challenging issues about the trust establishment between nodes, the enforcement of their cooperation and of the reliability of their backups, and the fairness of these various tasks. Achieving secure cooperation requires both the protection of data stored in the system, and the establishment of trust between participating nodes.

III. DATA PROTECTION

Data backed up in the system should be protected from disclosure and destruction performed by selfish or malicious nodes.

A.Data disclosure

The data being handed to unknown parties need to be protected against public disclosure. This can be enforced through the *encryption* of the data with a key that can be retrieved if the device is lost (e.g., passphrase or key escrow), together with data *fragmentation* and *dissemination* (i.e. the different data fragments of one file will be distributed to separate mobile nodes). This requires adequate key management mechanisms.

B.Data destruction

The data stored are exposed to the potential misbehavior of their temporary holders. The verification that one such holder still possesses these data after some time has elapsed (*data possession*) is the basis for the estimation of the data availability and therefore represents a necessary primitive to be able to react to data backup destruction. The verification task may be undertaken by another trusted node acting as a delegate for the owner. The availability of some data can also be increased using *replication*.

Protocols for data possession verification are generally based on challenge-response messages. We introduce a new protocol that allows a node to probabilistically verify against an encrypted memory whether a data holder still possesses the data he agreed to store for the originator. This protocol does not require the verifier to keep data or pre-computed challenges nor the prover to perform time-consuming computations to answer challenges. The proposed protocol comprises two phases: a storage phase and a verification phase (figure 1).

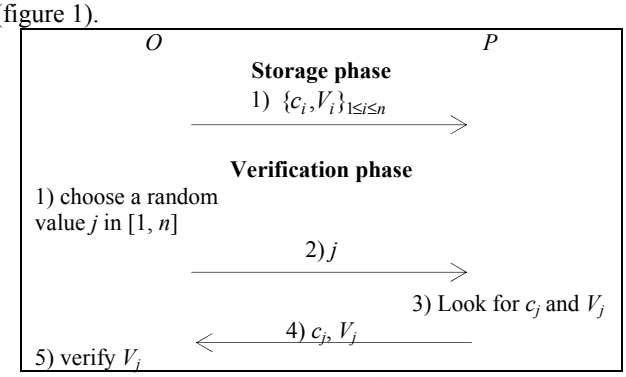

Figure 1 Probabilistic verification protocol

The set $\{c_i\}_{1 \le i \le n}$ corresponds to segments of the encrypted data. The set $\{V_i = f_{Ko}(c_i)\}_{1 \le i \le n}$ allows the integrity verification of data segments. The function f_{Ko} can be an encryption method (e.g., DES) or a one way hash function (e.g., MAC).

Performing c challenges allows the probabilistic detection of misbehavior. This value c is selected based on the reputation of the prover (that is, given by the portion of data deleted by this latter, which is k/n), and the probability of detection of misbehavior, which should be high enough depending on the criticality of the data backed up.

$$c = \lceil \log_{1-k/n}(1 - p_{detection}) \rceil$$

IV. TRUST ESTABLISHMENT

Cooperation between nodes should be evaluated on both long and short-term because contrary to packet forwarding for instance, evaluating the correctness of a backup cannot be completely immediate.

A. Long-term

The long-term evaluation of trust aims at encouraging the durability of nodes' cooperative behavior and at ensuring an effective backup. The evaluation may be implemented based on a reputation mechanism. The initial assessment of how trusted one may consider a node is based on the reputation self-carried by that node and updated by an authority (or a set of authorities) every time the node reconnects to a supporting infrastructure. The authority deterministically assigns a new reputation rating after checking if the node delivered the data it had promised to store. Nodes are forced to regularly contact the authority in order to renew the validity of their reputation as certified by the authority itself.

B. Short-term

The short-term evaluation aims at inciting nodes to accept storing data and at stimulating cooperation of nodes with similar mobility patterns (in particular in order to prevent failures due to backups becoming unreachable too fast). The evaluation of short term cooperation may rely upon two different mechanisms:

- Local reputation: is an estimate of how long-term reputation should be modified based on verification challenges directed towards data holders.
- Remuneration: is an estimate of the cooperative actions undertaken by the mobile node (e.g. verification challenges towards other nodes, storages accepted). Remuneration may be based on an optimistic fair exchange protocol, whereby an authority may address litigations a posteriori, when mobile devices are connecting with the supporting infrastructure again. Offline tamper-resistant hardware may also help resolve remuneration related litigations.

V. A SYSTEM PROTOTYPE: MoSAIC

MoSAIC[1] addresses data backup in the particular context of one-hop mobile ad hoc networks, which may be supported by opportunistic and transient pairing and communication with Bluetooth or 802.11. The backup application addressed in this project aims at enabling a mobile node to back up its data by exploiting the storage space of nearby accessible nodes through spontaneous interactions. It is also assumed that nodes may connect to an infrastructure from time to time, even though permanent connection would be too expensive for backing up non critical data for instance. Data could be transferred either immediately back to their owner if the mobile nodes are still in touch, or possibly through a trusted third party, for instance residing within the fixed infrastructure to which devices are regularly reconnected.

VI. CONCLUSION

Different approaches can be taken for cooperation enforcement. These approaches are dependent on application-specific requirements (e.g., self-organization) and environment characteristics (e.g., mobility). We now plan to validate the trust establishment process in terms of resilience to attacks, performance, and fairness using game theory, simulation, as well as prototype-based evaluation techniques. More evolved protocols for the verification of possession are also under investigation.

[1] The MoSAIC Project, project partners: Institut Eurécom, IRISA, LAAS. http://www.laas.fr/mosaic/