# Byzantine 2f+1 State Machine Replication with COTS Components

Giuliana Santos Veronese [1], Miguel Correia [1], Lau Cheuk Lung [2]
1 Universidade de Lisboa, Faculdade de Ciências, LaSIGE
2 Universidade Federal de Santa Catarina

## Introduction

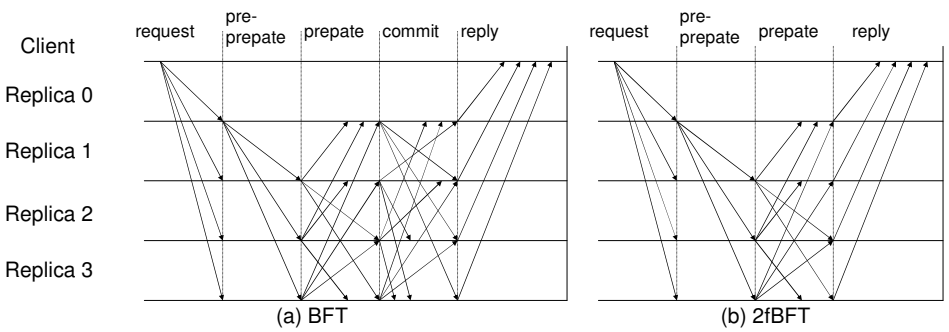
The number of malicious attacks has been constantly growing making computing security as a whole an important research challenge. To meet this challenge several Byzantine fault-tolerant algorithms have been proposed. Byzantine fault-tolerant systems are usually built using replication techniques and ensuring a set of safety properties.

Reducing the **number of replicas** has a very important impact on the cost of the system.

✓We propose a novel Byzantine fault-tolerant **2 f+1** state machine replication algorithm (**2fBFT**) that uses only COTS components

✓The algorithm is based on the **Trusted Platform Module** (TPM) designed by the Trusted Computing Group, currently shipping as a chip in commodity PCs

## Protocol




(a) BFT (b) 2fBFT

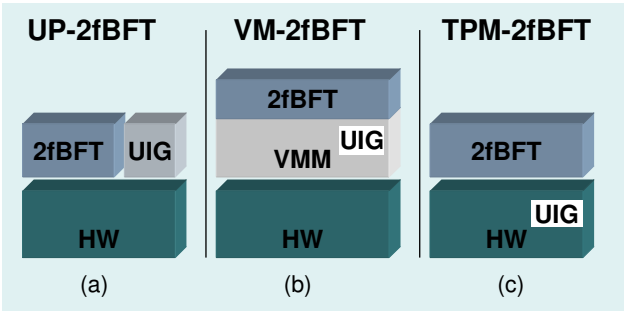
✓A service called *Unrepeatable Identifier Generator (UIG)* assigns a unique identifier to a message using standard TPM commands

✓This identifier is verifiable, i.e., anyone that has access to the sender's public key can check if it is valid. This prevents a malicious replica from sending the same message with different identifiers to different replicas

✓This service is used to implement a *Confirmable Reliable Multicast (CRM)* abstraction, which delivers the same messages to all replicas and allows the replicas to prove that they delivered a message

✓CRM provides similar functionality to BFT's pre-prepare, prepare and commit phases

## Implementation Scenarios




UIG implemented as a user-level process (a), in a trusted VM (b) and using the TPM (c)

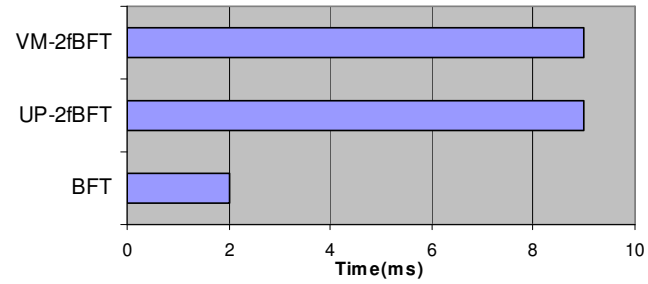
✓The prototype was implemented in Java. UP-2fBFT and VM-2fBFT do signatures with NTT ESIGN (2048-bit keys)

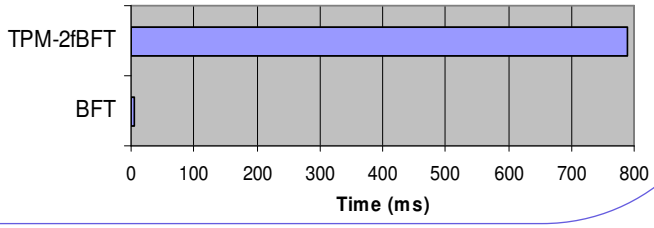
✓In the experiments we considered at most one faulty replica (f=1), requiring 4 replicas for BFT and 3 replicas for 2fBFT

✓The replicas and client nodes were 2.8GHz Pentium 4 with 2GB RAM and were connected over a Fast Ethernet 100Mbps

✓ The experiments measure the latency to invoke a *null* operation. The response time is measured at the client node

## Experimental Evaluation




This work was supported by the EC, through project IST-4-027513-STP (CRUTIAL) and Alban scholarship E05D057126BR, and by the FCT through the Multiannual and the CMU-Portugal Programmes.

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DE LISBOA