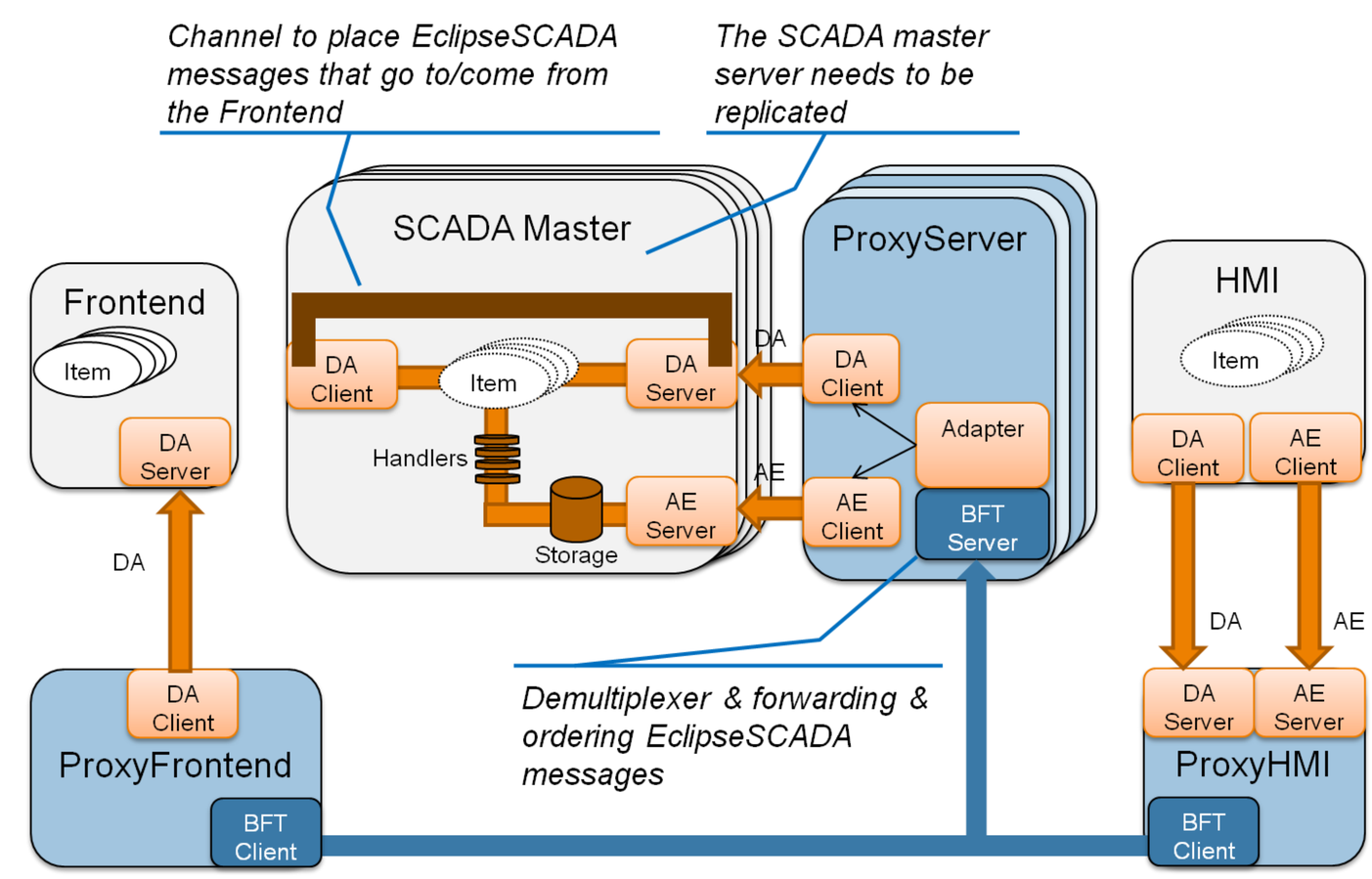
Intrusion-Tolerant SCADA System

Contact: Nuno Neves, Alysson Bessani
{nuno, bessani}@di.fc.ul.pt
FCiências.ID / LASIGE, Faculdade de Ciências, Universidade de Lisboa
www.navigators.di.fc.ul.pt

Overview

Supervisory Control and Data Acquisition (SCADA) systems form the backbone of critical infrastructures, including the power grid. One of the major threads is if an attacker gains access to the main computer – the SCADA Master – as it manages all operations under supervision, and therefore can lead to a catastrophic scenario.

We have built an intrusion-tolerant SCADA system, where the SCADA Master is replicated, as a solution to enhance the existing protection capabilities against malicious attacks. Our prototype is based on the open-source EclipseSCADA system, which was integrated with the BFT-SMaRt library to manage the replicas. Initial experiments show the potential of the solution, as both accidental failures and attacks could be tolerated under acceptable levels of performance.



Challenges

Main challenges in integrating a Byzantine fault tolerant (BFT) replication library with the Master server of Eclipse SCADA:

1) **Multiple communications**. The Master encompasses several communication entry points to interact with the Frontends and HMI. Consequently, the Master replicas can receive requests and replies from all these modules in some random order.
2) **Concurrency in processing.** Internally, the Master server does not operate sequentially. The DA and AE subsystems have several modules that execute concurrently with multiple threads, enabling requests to be processed in parallel in some random order.
3) **Timestamps.** When an event is created, a timestamp is retrieved from the operating system and assigned to the event. It is necessary to ensure that all replicas generate the same timestamp for the same event to avoid the violation of determinism.
4) **Asynchrony**. The Master server can send to the HMI multiple messages in response to a single event reported by a frontend. As the HMI receives messages asynchronously from a set of replicas, the messages must contain some sort of information that enables it to understand in which context that messages were produced.
5) **Performance**. Some critical infrastructures use Eclipse SCADA nowadays, confirming that its performance is suitable for deployments in the field. In a replicated version of the Master server, the performance may be affected due to the overhead introduced by the intrusion-tolerant library.

Approach

Modern SCADA systems normally employ fault tolerance techniques to ensure system reliability. The SCADA Master is often deployed in a hot-standby configuration, where a backup can take over the primary server in case a failure is detected. These approaches, however, cannot address the compromise of the Master.

Ideally, the SCADA Master should operate correctly even in presence of malicious faults. By resorting to appropriate replication techniques, the SCADA Master could be enhanced to operate correctly not only in the presence of accidental faults but also when there are compromises. This new intrusion-tolerant SCADA would contain improved protection capabilities against malicious attacks, complementing more traditional security solutions.

We have developed such an intrusion-tolerant approach and applied it to the open-source Eclipse SCADA project. Generically, we resort to State Machine Replication (SMR) to manage multiple replicas of the SCADA Master. The key idea is to make replicas execute deterministically the same sequence of requests in such a way that, despite the failure of a fraction of the replicas, the remaining ones have the same state and ensure correctness of the offered services through a voting on their responses. The protocols were built to address arbitrary (or Byzantine) failures, which include ordinary problems like crashes but also (compromised) replicas that act maliciously.

Experiments

Research questions:

1. What is the cost of sending ALARMS from the Frontends to the HMI? *(see Fig 1)*
2. What is the penalty of UPDATING variables from the HMI to the Frontend? *(see Fig 2)*

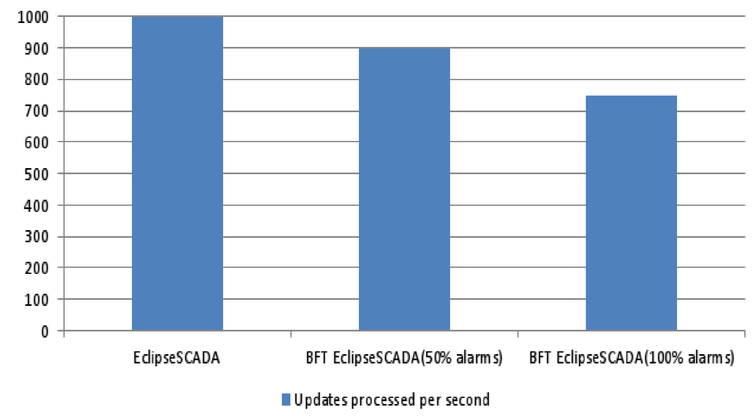


Figure 1. Transmission of alarms from the Frontend to HMI.

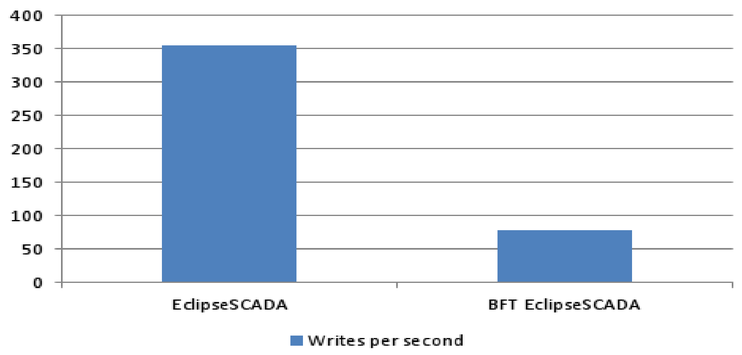


Figure 2. Updating items in the Frontend from the HMI.

References

[1] André Nogueira, Alysson Bessani, Nuno Neves, *Intrusion-Tolerant Eclipse SCADA*, Symposium on Innovative Smart Grid Cybersecurity Solutions, March 2017

SEGRID Security for smart Electricity GRIDs
www.segrid.eu info@segrid.eu

This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 607109