# SIMULATING A SOFTWARE REPUTATION SYSTEM WITH MALICIOUS USERS

ANTON BORG
BTH

Today, computer users have great trouble in separating malicious programs from legitimate software. Traditional countermeasures such as anti-virus tools protect against truly illegal and malicious programs, but the situation is complicated by a ”grey-zone” of questionable programs that exists between these malicious programs and legitimate software. The programs in this grey-zone are hard for different protection programs to handle and almost impossible for a single user to judge. We therefore suggest a software reputation system to help computer users in separating legitimate software from its counterparts.

In order to evaluate our proposed system we have implemented a simulator [1]. Using this simulator, a balanced, well-informed rating of judged programs appears, i.e. a balance between quickly reaching a well-informed decision and not giving a single voter too much impact. More recently, our simulator also implements the ability to simulate malicious users. The malicious users are designed to be sleepers, i.e. they will appear to be normal users while building up their trust factor in the community before turning malicious. We have found that an attack by malicious users will not affect the overall system, unless the number of attackers are unrealistically high. However, a coordinated attack from a limited amount of attackers against a selected subset of the applications may distort the reputation of these applications.

We have found that malicious users may be defeated or identified either by limiting the number of votes given or logging the extra amount of votes given to program under attack. In addition, resetting the trust factors that specify the users impact decreases the effect of malicious users. It can be argued that since users will not be aware of their trust factor at any given moment, a function that randomly selects users and resets their trust factor can be implemented without damaging the trust user have towards the system.

## References

[1] M. Boldt, A. Borg, and B. Carlsson. On the simulation of a software reputation system. *Availability, Reliability, and Security, 2010. ARES '10 International Conference on*, pages 333 – 340, 2010.