TRUSTING A COMPUTER WITH YOUR MONEY

Gordon J. Pace

Dependability and Reliability in Security-Intensive Financial Systems is a two year project funded by the Malta Council for Science and Technology through the Malta National Research and Innovation (R&I) Programme 2008

EDITOR'S NOTE *Gordon J. Pace is an Associate Professor with the Department of Computer Science at the University of Malta. His research interests lie primarily in techniques for the development of safe and dependable software and hardware systems.*

It is becoming increasingly rare to engage in any activity which does not involve direct or behind-the-scene interaction with computers. Whether you are driving a car, using a washing machine, booking a theatre ticket or listening to music, machines are crunching information in the background. Despite popular perception, dependability of your software does not just concern developers of nuclear power station controllers, medical software and autopilot systems. Consider the development of software handling financial transactions. Errors in the software can easily lead to huge financial losses: directly through miscalculations, or indirectly by not preventing misuse of the system.

"Errors in the software can easily lead to huge financial losses"

Over the past two years, the University of Malta together with Ixaris Ltd Malta were involved in a project on increasing the dependability of software, focusing in particular on financial systems. Ixaris Ltd provides services for their clients to deposit funds through funding instruments (such as their own personal credit card or through a bank transfer mechanism) and spend such funds through spending instruments (such as a virtual Visa card or a plastic MasterCard). The service is used worldwide and thousands of transactions are processed on a daily basis.

This software, as with most commercial systems, has been thoroughly tested, and has various internal checks for consistency of data. Testing, corresponding to an accountant having to take exams before being allowed to practice, and internal checking within the software itself, corresponding to an accountant cross-checking himself or herself, can go a long way, and will only fail in exceptional circumstances. Still, those exceptional circumstances can be too costly to accept.

An increasingly popular approach adopted for improved dependability is that of runtime monitoring. In real life, this approach would roughly correspond to an independent auditor who, at the end of the year, checks the accounts for any inaccuracies, and certifies them correct if none are found. In a software setting, the system would simply be modified to create an audit trail – a log of relevant information. Another software system would be checking this trail against the business logic flagging any violations. Unlike traditional auditing, which is performed a posteriori – after the accounts have been drawn up – in runtime monitoring any violations can be caught immediately, and possibly remedied before the system proceeds. For example, if an error in the system would allow a client to transfer excessive funds, despite the fact that this goes against the terms and conditions, the transaction can be stopped and the parties concerned notified.

Still, there is no such thing as a free lunch, and online monitoring comes at a cost of system resources. If every system action has to be checked, the system has to stop until the verification is complete before proceeding further. Slowing the system down can lead to downgraded performance and hence dampen the clients' user-experience. This is where the research project comes in. We have developed a novel monitoring approach, in which the system is monitored on a separate machine. The monitor tries to keep up with the system, but never impedes it from proceeding, thus having no adverse impact on its performance. However, if a problem is encountered, the transaction being processed is temporarily stopped and any actions which may have been performed since the unexpected action are, if possible, reversed.

"Online monitoring comes at a cost of system resources"

Using this approach, the dependability of the system is substantially increased without adversely affecting the system performance. Dependability is increasingly becoming a leading issue in software development. Traditional approaches to testing can no longer cope with the increased size and complexity of the systems, and new approaches have to be developed. Peace of mind about your money is no longer merely about physical and financial trust in whoever handles your financial transactions. You must also trust their software.