

Comments to the European Commission on the Draft Final Report on Availability and Robustness of Electronic Communications Infrastructures

Submitted by
Information Society Strategy Working Group of the
Green League (Finland)
tyst@greens.fi

This document summarises the comments made on the draft final report¹ on 'Availability and Robustness of Electronic Communications Infrastructures', as invited by the European Commission². These comments have been prepared by the Information Society Strategy Working Group of the Green League (Finnish Green party, <http://www.greens.fi/>).

Comments to key recommendation 6

Key recommendation 6 addresses the issue of software supply chains, and states that there is an "increased risk brought through dependency on software-controlled technology". The risk has been correctly identified, but we feel that several critical building blocks are missing from the recommendation. Detailed comments on these issues follow.

Risk of single actor domination. In order to actually meet the requirement "providing hardware and software supply chain technology and assurances for integrity regardless of where or by whom, the technology was designed, developed, manufactured, or deployed", as articulated in the key recommendation, it is necessary to have a technical requirement of using open and standardised interfaces between software (and hardware) components. Openness in this context does not only mean the openness of specifications, but also freedom of choosing the software vendor without being tied by intellectual property rights of another vendor.

Unless openness (both in IPR considerations and specifications) of component interfaces of communications networks is ensured, a single actor in the marketplace can still control the supply chains that are being used.

In a crisis, the use of a single actor may cause problems for various reasons, such as not being able to meet their service level agreements due to the high volume of support requests, or because the actor may reside in a country from which support cannot be obtained due to economic or military reasons, or in an area affected by a natural disaster. To draw an analogy to positioning systems, augmenting the GPS system with Galileo system is partly addressing a similar threat.

Risk of non-access to source code. During a prolonged crisis, software supply chains may need to be revisited. For example, some vendors that normally provide services may be unavailable for

1 http://ec.europa.eu/information_society/newsroom/cf/document.cfm?action=display&doc_id=290

2 http://ec.europa.eu/information_society/newsroom/cf/itemdetail.cfm?item_id=3334

use for various reasons, especially at times when the crisis has an economic/political or military aspect to it. In these cases, it is important to be able to switch vendors, or at least to be able to obtain fixes and corrections to software from other parties than the original vendor.

If the software has been developed under a closed-source model, where even the customer has no access to source code, it is impossible to switch the vendor or make corrections even in the case where the customer would have suitable expertise. In many closed-source models, the companies use the services of so-called escrow companies, that are supposed to release the source code in specific circumstances. However, in a crisis, escrow companies' role cannot be trusted as they may also be under similar restrictions. In addition, source code escrow is typically available only to large organisations; small customers usually do not have the bargaining power to demand source code escrow at all. Therefore, the only real way to guarantee that the user of software has access to source code in a major crisis is to place the source code in hands of the user from the start.

Therefore it is imperative to favour open-source development for critical communications infrastructure. In this case, open source does not necessarily need to be free software (software libre), but the important aspect is that the actual source and the code development environment are available to the customer of the software. In this way, critical and 'ad hoc' corrections can be made to the systems without consulting the original vendor.

General comments on the report

It seems that the report has a bias against open source. This is evident in, for example, footnote 75, which refers to a study in which (the authors claim) it is shown that 'open source *negatively impacts reliability and security*' (emphasis added). Curiously, this is the only place in the whole document where open source is mentioned.

The reference seems to be erroneous: According to the referenced proceedings³, the "subject matter experts" (meaning workshop participants, many of which come from the same companies), the participants voted on whether "Open Source Software contributes to *better reliability and security*" (emphasis added). According to slide 16 of the proceedings, 77% of the participants voted "yes" - exactly the opposite of the claim in the report.

On the other hand, the referenced proceedings actually argue in support of Open Source by mentioning monopolistic positions of software vendors on slide 19 (monopolies in Open Source would be very hard to pull off). Also, the proceedings state that "When using third party components, it is difficult to determine what security standards they are following" - also something alleviated by access to source code.

We believe that ignoring Open Source and the benefits that it can offer to critical infrastructure would be a grave mistake. Instead of having a negative effect on security, open source actually has a major positive effect. In addition to the ability for a customer to make corrections to software by themselves (as mentioned previously), open source enables the customer to actually verify that security fixes have taken place; it provides a way to verify that supply chain vendors have not added backdoors or Trojan horses in the code; it facilitates third-party security analysis much better; and finally, it removes the possibility of selling "security through obscurity", a practice which is still widespread in many software engineering companies.

The report ignores threats of software monoculture. Software vulnerabilities share a similar trait with biological viruses and contagious agents: their effect is worst on populations that are homogenous. As an example from biology, the commercial Cavendish banana plant which

³ <http://www.comsoc.org/~cqr/Docs/Events/EU-Workshop/W3%20Issues%20voting.pdf>

provides most of world's current banana supply is genetically homogenous - all plants are clones. In addition, the plant is often grown in a monoculture. There is a reason why Cavendish is the current crop: the previous main export banana crop, Gros Michel, was wiped out by a fungus that devastated that homogenous banana monoculture. There is no real reason why it could not happen again to Cavendish, and this risk was actually reported in mainstream media in 2005-2006.

Software is similar to bananas in the way that if a sufficiently large portion of devices (be they end-user devices or network devices) run the same software, the same type of vulnerability may take out a large number of devices.

The activities towards a more robust communications network should therefore guarantee that different vendors' software are used; not all components are running on a single type of operating system or platform; and that there are always alternative routes which are being implemented using a different hardware/software combination.

The report does not specifically address information systems warfare. Information systems warfare is an existing threat. It differs from conventional warfare for example in that it does not necessarily require major investments in military technology. Attacks can be mounted by small groups and as such, information systems warfare lends itself well for asymmetric and guerrilla warfare, especially when the attacking group is using the same infrastructure services as the target (the attacker cannot be defended against "at the border").

Information systems warfare attacks may use malware (viruses, Trojan horses, etc.) as the mode of attack. The target may be information (extracted from governmental, commercial or military systems), or to trigger a cascade failure by targetting an infrastructure system such as electricity grid or municipal water systems, or even as a part of psychological warfare by targetting supermarket cashier systems, which would definitely interfere with the day-to-day lives of most people.

The recommendations above (avoiding a single dominant vendor in a supply chain, avoiding monoculture, and promoting the use of open source) help against any generic security threat, which also includes information systems warfare. However, the European Union should actively promote systematic vulnerability and robustness testing and hardening of existing systems, support technical, leading-edge vulnerability research in university level and in independent software companies, and make sure that all member countries have a sufficiently well resourced CERT (Computer Emergency Response Team) unit. CERTs should specifically aim to support SMEs and private individuals to protect their systems, as those parties may not have resources to run their own IT security departments.