Security of bespoke solutions vs academic curricula

Maciej Miłostan[1),2)*], Mikołaj Dobski[1)*], Paweł Berus[1)], Gerard Frankowski[1) *]

1) Poznań Supercomputing and Networking Center, IBCH PAS, Poznań, Poland

2) Poznań University of Technology, Institute of Computing Science, Poznań, Poland

In the presentation, we will share select experiences and observations collected throughout more than 20 years of code and infrastructure audits, pen-tests and teaching activities. We want to describe our best practices while improving the security of bespoke software solutions developed at PSNC and our partners within the GÉANT community – usually under tight time and resource constraints. Then, we will summarise the experience gathered conducting classes for computer science students at PUT (Poznan University of Technology) and executing over a dozen Secure Coding Training (SCT) events for GÉANT.

We will show why the Secure Development Life Cycle is important and how we can foster its adoption among junior staff, especially developers and, hopefully, senior management.

We will identify the gaps in the education of young generations and show how we can improve University curricula to match market expectations and arm the graduates with a more realistic perception of the contemporary threat landscape. Did you know you could even get a Computer Science degree at some renowned universities without attending any cybersecurity-related course? It was the ground truth recently, not only at the beginning of the Digital Dark Ages.

A decade ago, while teaching students, we focused mainly on the principles of cryptography (the classical one) and data protection. However, during that time, data protection was more oriented towards hardware failure protection mechanisms (RAIDs, etc.) than legislation and procedural stuff.

Nowadays, it is necessary to review and update existing curricula to include principles of code audits such as code reviews as well as treatment of SAST and DAST tools as mandatory in software engineering. Young developers need to know about typical bad smells in the software, sometimes leading to security vulnerabilities. It is also essential to teach students about real-life vulnerabilities and methods of identifying vulnerable services in the network (and avoiding introducing vulnerabilities in the developed source code).

The authors are engaging with local universities and the GÉANT community, on the one hand, to fill the educational gap and, on the other hand, to provide complimentary code and system auditing services, striving to improve the overall security of services deployed within the GÉANT network and project partners – especially NREN operators.