Homework lecture 4
Economics of network security

Harald Vranken
harald.vranken@ou.nl
February 17, 2020

Question 1
The effort that users have to take for following security advices, is often ignored. Why does this lead to a ‘tragedy of the commons’?
(Hint: see section 7.3 in Herley’s paper.)

Answer
The follow up of security advices requires the user's attention and effort, which are limited and imply implicit costs. Each individual security advice may have benefit, but all together these advices imply a cumulative burden for all users. Just as villagers will overgraze a commonly held pasture, security advices demand more effort than users can provide.

Question 2
Explain why ‘CVSS is DoS-ing your patching’.

Answer
The number of vulnerabilities reported is huge (on average nearly 60 per day according to VulnDB). There are incentives, both for security researchers and security suppliers, to assign high CVSS score to reported vulnerabilities. Hence, users are requested to install large numbers of patches to repair critical vulnerabilities on a daily basis. In practice, this is unfeasible and hence users will not do so. Furthermore, many high-rated vulnerabilities are never actually exploited, which further decreases the willingness of users to patch.

Question 3
Why can the market for secure software be considered as a ‘market for lemons’?

Answer
Security vendors may assert that their software is secure, but users cannot measure this. Hence, there is asymmetric information between vendors and users. This leads to low-quality products, since user are unwilling to pay higher prices for secure software, while vendors are unwillingly to sell secure software at low prices and become disinclined to invest in security measures.

Question 4
Would moving liability to software developers (eg. by removing all disclaimers on liability for defects from contracts) offer a way to enhance software security?
(Note: we did not talk about this in the lecture, but give it some thoughts!)

Answer

This may cause software developers to put more effort on security. However, it offers no panacea. Introducing software liability increases the risk for software developers of being sued. This may reduce the pace of innovation, both for commercial companies and for contributors to the free software community. Second, given the state-of-the-art in software engineering, it is unrealistic to assume that software can be produced that is guaranteed to be free of vulnerabilities.