Software Reliability and Security: Supply Side Complements (in progress)

- Peter Honeyman & Galina Schwartz

center for information technology integration

problem statement

- we study manufacturer incentives to invest in software reliability and security (R&S)
- what factors determine these incentives?
- expected profit depends
  - technology and information structure
  - legal framework (property rights and their enforcement)
  - market structure
  - consumer (user) demand
- …and why do we care?
  - software is a public good, which results in
  - negative production externalities of R&S

externalities in software production

- externalities in provision of software R & S
  - production side (manufacturer supply is suboptimal)
  - consumption side (user demand is suboptimal)
  - effects of hackers on software R & S
- suboptimal manufacturer incentives due to
  - software is a public good [see Varian (2002)]
  - information deficiencies [folklore + prevailing view of the literature]
  - legal (+ enforcement) deficiencies [folklore + prevailing view of the literature]
- we focus on manufacturer incentives
  - we do not address the effects of non-optimal user demand, [see Honeyman and Schwartz (in progress)]
  - we assume exogenous user demand

summary of causes of externalities

- public good → free riding problem (s)
- information deficiency
- manufacturer heterogeneity
- legal deficiencies
  - poorly defined & conflicting property rights
  - high enforcement costs

software externalities: a closer look

- software products of different manufacturers are consumed concurrently [free riding problem (s)]
- information deficiency prevents to determine the failed product exactly
- software R&S (both: reliability and security) have the weakest-link prototype technology
- the weakest-link means that composite product's R or S is equal to R or S of the product which has the lowest R or S

players

- whose incentives matter?
- 1. Manufacturers
- 2. Consumers / Users
  - average /regular users
  - system administrators
  - security specialists
- 3. Hackers - users, whose objectives DIFFER from usual user objectives

I. software R&S: production technology differs

- improving R&S: technical means differ
  - to improve reliability: debug & test extensively
  - to improve security: debug & test even more
- → production technology: which problem is harder R or S?
  - reliability: costly, but routine task [science?]
  - security: costly, not routine task [art?]

II. R&S: manufacturer incentive to invest

- improving R&S : economic incentives concur
- R&S software: production complements
- Reasons of supply side complementarity
- user actions in troubleshooting R&S are identical
  - how users troubleshoot reliability & security problems
- user actions:
  - find a fix, download and apply it
  - Perform full /complete system reinstall
- ➔ manufacturer incentives to improve software R&S concur because users do not discriminate between R&S problems

III. production of software R&S

- Summary: manufacturing software R&S

to improve software reliability and security

- technical means differ (see I)
- economic incentives concur (see II)
- Implications:
  - R&S are complements in production
  - treating investments in R&S as substitutes ➔ yields production inefficiencies

the literature & the players

- the existing literature
  - mostly focuses on manufacturer incentives example: a concise summary by Hal Varian (2002)
- we consider incentives of other players [hackers]
- surprisingly: hacker presence results in externality of reliability on security [R on S]
- we are inconclusive about the sign of this externality: it depends on hacker objectives and R & S levels
- But
  - if R & S levels are high, externality is positive
  - if enforcement costs are prohibitive, externality of R on S may be negative!

users: regular users and sysads

- user responses to software failures
  - do not differentiate if failure reliability or security driven (the task is too cumbersome)
  - Same response to any software failures:
    - find a fix and download it
    - reinstall the system from scratch
- what users of all types have in common?
  - only occasionally tune the system reliability and security
  - interest in reliability & security is dormant until major loss (financial or informational)

user demand → manufacturer incentives

- manufacturer incentives to invest in R&S
- are complements [supply side complements]

- **main result:**
- **software reliability and security are production complements**

users actions and R&S failures

- users do not discriminate between R&S
- why? because for users R&S failures are undistinguishable
  - user respond to reliability and security failures IDENTICALLY
- evidence:
  - Song & al. (2001)
  - your own actions when your computer fails

hacker types

- hacker types
  - White Hat
  - Grey Hat
  - Black Hat
  - Other ?

hackers: differences and similarities

- hackers differ by / in [and differ a lot!]
  - resources [means]
  - expertise [means]
  - reasons to hack
- Q: what do all hackers have in common?
- A: All hackers dislike being persecuted (punished / arrested / imprisoned)
- ➔ all hackers are similar in
  - avoidance of being caught
  - ➔ all hide their penetration / presence

hackers: means of concealment

- hackers: hide penetration and presence via:
- alteration of the system (files and directories):
  - changing, renaming, erasing, adding
- to destroy the evidence (changing and erasing)
- to hide the presence (all ...ings)

hacked system and buggy system

- from user perspective
- hacked system and buggy system behave similarly
- user actions are identical because
  - it is too costly to discriminate between S&R
- from user perspective:
- hacked system ≈ buggy system

hacker incentives to hack

- Statement:
- reliability ↑ ⇔ incentives to hack ↓
- why incentives to hack a more reliable system are lower?
- Or: improved software reliability lowers incentives to hack

positive externality?

- now the externality is tiny. we expect its increase
- improved reliability improves system security
- Because
  - reliability ↑ ←→ incentives to hack ↓
- software more reliable → system more secure [reduced incentives to hack]
- positive externality of improving software R on S [due to disincentives to hack]
- this reinforces our result of production complementarity

thank you for your attention

http://www.citi.umich.edu/