Course outline

1. Language-Based Security: motivation
2. Language-Based Information-Flow Security: the big picture
3. Dimensions and principles of declassification
4. Dynamic vs. static security enforcement
5. Tracking information flow in web applications
6. Information-flow challenge

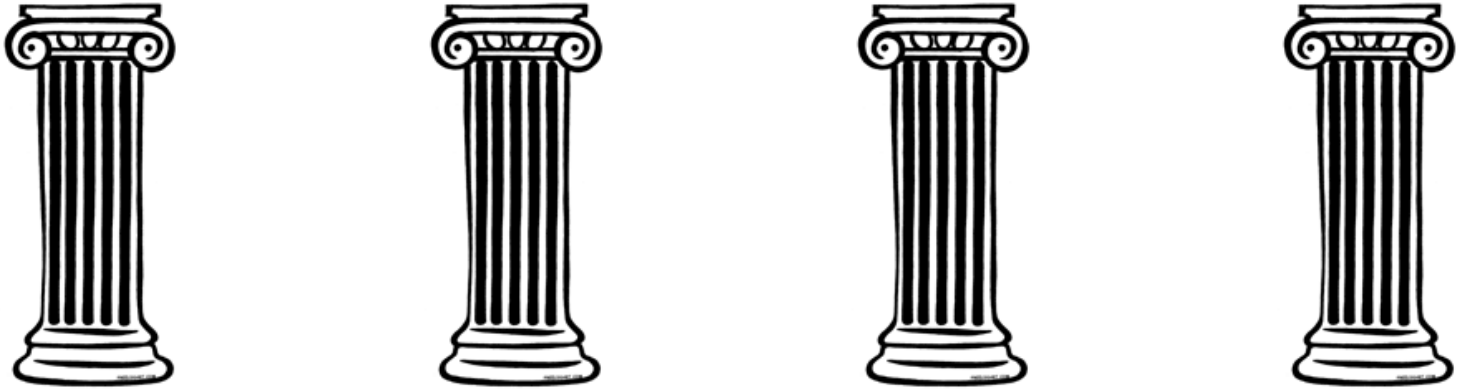
Web: foundation of modern society

E-commerce

Healthcare

Government

Individuals



The Web

Web: foundation of modern society

Attacks against web applications: 60% of Internet attacks

E-commerce Healthcare Government Individuals

sensitive data exposure

cross-site request forgery

cross-site scripting

session missmanagment

insecure references

broken authentication

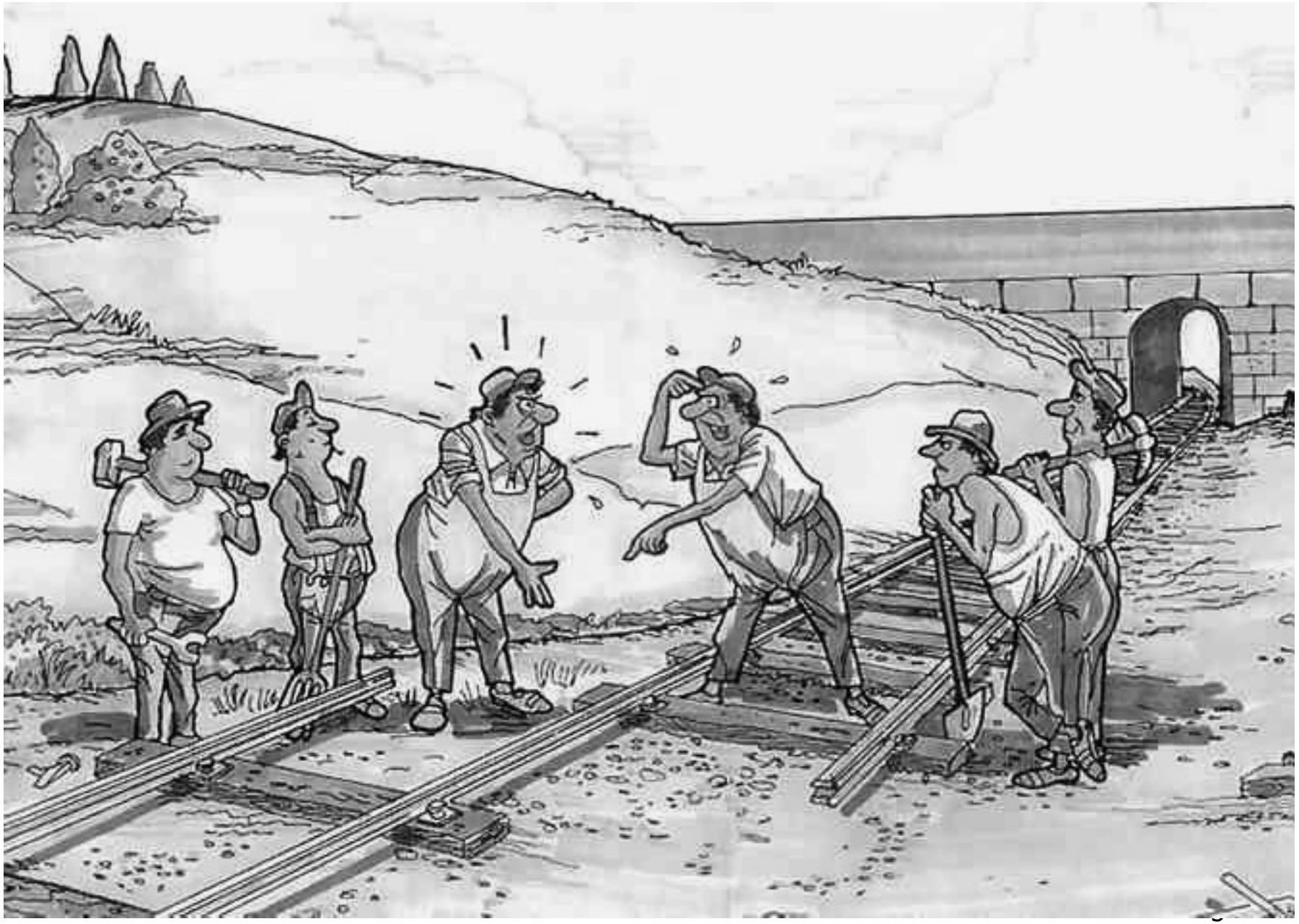
injection

security misconfiguration

flawed access control

Web security challenge

- Web not designed to be secure
  - Static pages to be read by scientists (CERN 1990)
  - stateless HTTP protocol
- Moving target ever since
  - Features "bolted on"
- Tremendous evolution
  - JavaScript
  - dynamic web 2.0
  - stateful HTML5
- Security cannot be "bolted on"



Student Directions for the

https://student.portal.chalmers.se/en/contactservice/ITServices/Pages/StudentDirectionsfortheUseoftheITresourcesatChalmers.aspx

Resources at Chalmers

Passwords may not be disclosed to other students or individuals.
If you're compelled to write down your passwords and log-in information, please keep this information away from and outside the reach of others and stay away from using plain language when referring to this type of sensitive information.
Never save your user name and password in the browser for automatic log-in regardless of which computer you use.
Passwords must be immediately changed if you suspect that someone else may have come across such sensitive information.
Passwords to access the IT resources of Chalmers may not be simultaneously used on other websites or systems.

Private PCs

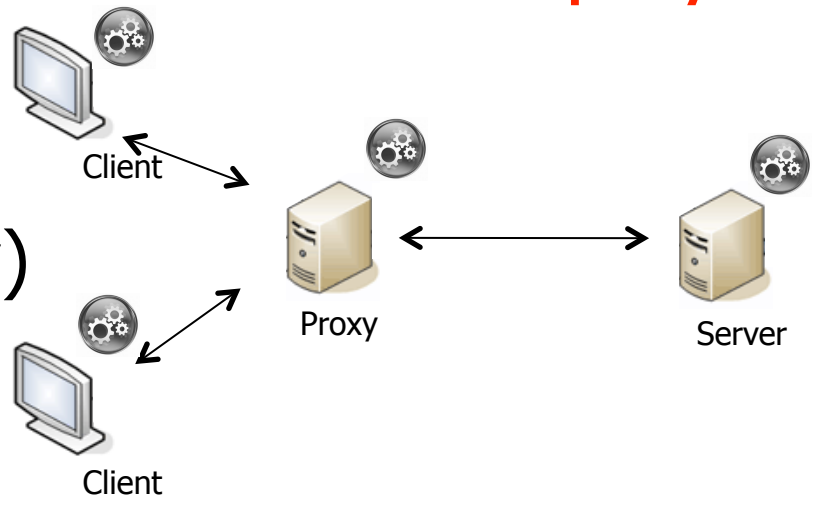
All computers and other equipment connected to the Chalmers data network, regardless of ownership and location, must have up-to-date antivirus protection and a current version of their operating system. Private PCs must always be checked to make sure they have not been contaminated with malware. All computer user names for the various departments must be equipped with secure passwords (please consult https://www.testalosenord.se/ in the case of uncertainty).

"please consult https://www.testalosenord.se in the case of uncertainty"

Demo

From JSFlow to architectures for inlining

- JSFlow: Tracking information flow in JavaScript
  - [Hedin & Sabelfeld'12]
- As security-enhanced JavaScript interpreter
  - [Hedin, Birgisson, Bello & Sabelfeld'14]
  - Software release: http://chalmerslbs.bitbucket.org/jsflow
- Implemented itself in JavaScript
- Any interpreter of this kind can be deployed
  - as browser plugin
  - as web proxy
  - as service (suffix proxy)
  - integrator-driven
  - [Magazinius, Hedin & Sabelfeld'14]



Password strength checking

Browser extension

Replace the JavaScript engine

Web proxy

.evaluate(

)

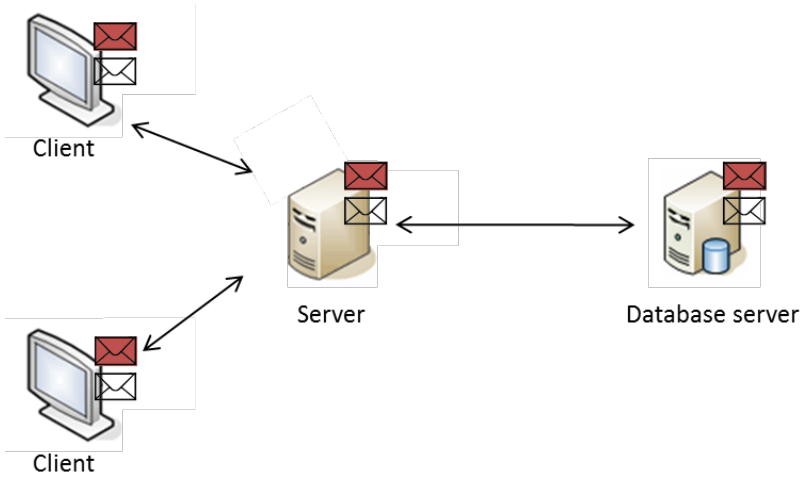
Web service (suffix proxy)

Integrator driven

Demo

Conclusions

- Web application security
  - moving target
  - attacks will flourish unless:
- Principled approach
  - security by intent
  - security by construction
  - information flow tracking
  - from foundational principles to practical security



Course summary

- Language-based security
  - from off-beat ideas to mainstream technology in just a few years
  - high potential for web-application security
- Declassification
  - dimensions and principles
  - combining dimensions key to security policies
- Enforcement
  - type-based for "traditional languages"
  - dynamic and hybrid for dynamic languages

Information flow challenge

- Attack the system to learn the secret
- Type systems to break
  1. No restriction
  2. Explicit flows
  3. Implicit flows
  4. Termination
  5. Declassification
  6. Exceptions
  7. Let
  8. Procedures
  9. References
  10. Arrays

http://ifc-challenge.appspot.com/

- Bonus: http://chalmerslbs.bitbucket.org/jsflow/jsflow-challenge.html