Dev and Blind

Attacking the weakest link in IT security

A Talk by Johannes Hofmann and Mario Heiderich
Confidence 201002, Prague

Introduction

- **Johannes Hofmann**
- Ten years of web development experience
- Security advocate
- Senior Developer at a major european social network
- @c_ion

- **Mario Heiderich**
- PHPIDS
- HTML5 Security Cheat Sheet
- Researcher for RUB and Microsoft
- Web Developer Background
- @0x6d6172696f

What's to come

- Why to attack developers
- A lot of love
- What's the „web dev dilemma“
- Attacks against developers
  - Be prepared to see scary things
  - New offensive techniques – so fresh they don't even have an acronym yet
- Protection mechanisms that really work
- Discussion and Q&A

Developers vs. Security folks

- It's not really love connecting us...

- Developers create **crappy** code
- Especially web developers do!
- It happens all the time, every day
- Devs don't have any clue about **anything**
- „Just get it to work and you'll be fine!“
- Security folks **hate** devs - despite bugs being their bread and butter

- Mocking developers couldn't be more fun!

Remember that Debian thing in 2008?

Love is in the Air

- Devs create tools for people to *use*, attackers just *abuse* the work of others
- Good developers get fired when you sec guys release 0days
- Many Sec Consultants show up for a few days, slap exploit lists in your face and disappear only to repeat the process next year
- Security people meet in shady places and don't share their ivory tower knowledge


The Developer's Lament

The Developer's Lament 1/2

- Security **hurts** performance
- Security **hurts** user acceptance
- Security **hurts** development time

- Even relative security is expensive
- Faulty security only becomes notable once it's **too late**

The Developer's Lament 2/2

- Execs believe that security can be bought (stern look at audience)
- Maybe a fancy IBM scanner license for 33K USD?
- Dependencies on third-party code and services make it hard to audit a system in it's entirety
- Can you audit a large 200+ table monster at all?
- Unreviewed legacy code is everywhere - but refactoring is not a priority
- The deadline reigns supreme

- And worst of all we're juicy attack targets ourselves...

Why is that?

- Developer machines are located in the "secure" Intranet
- Devs are usually allowed to have highest privileges on their workstations
- Developers are using web based tools and store tons of credentials in their browsers
- Developers are power users and use a wide variety of software
- Developers are well connected and easy to reach
  - Twitter, Facebook, Skype, GTalk, ICQ...
- They usually have unrestricted access to the extranet during work hours

What to attack?

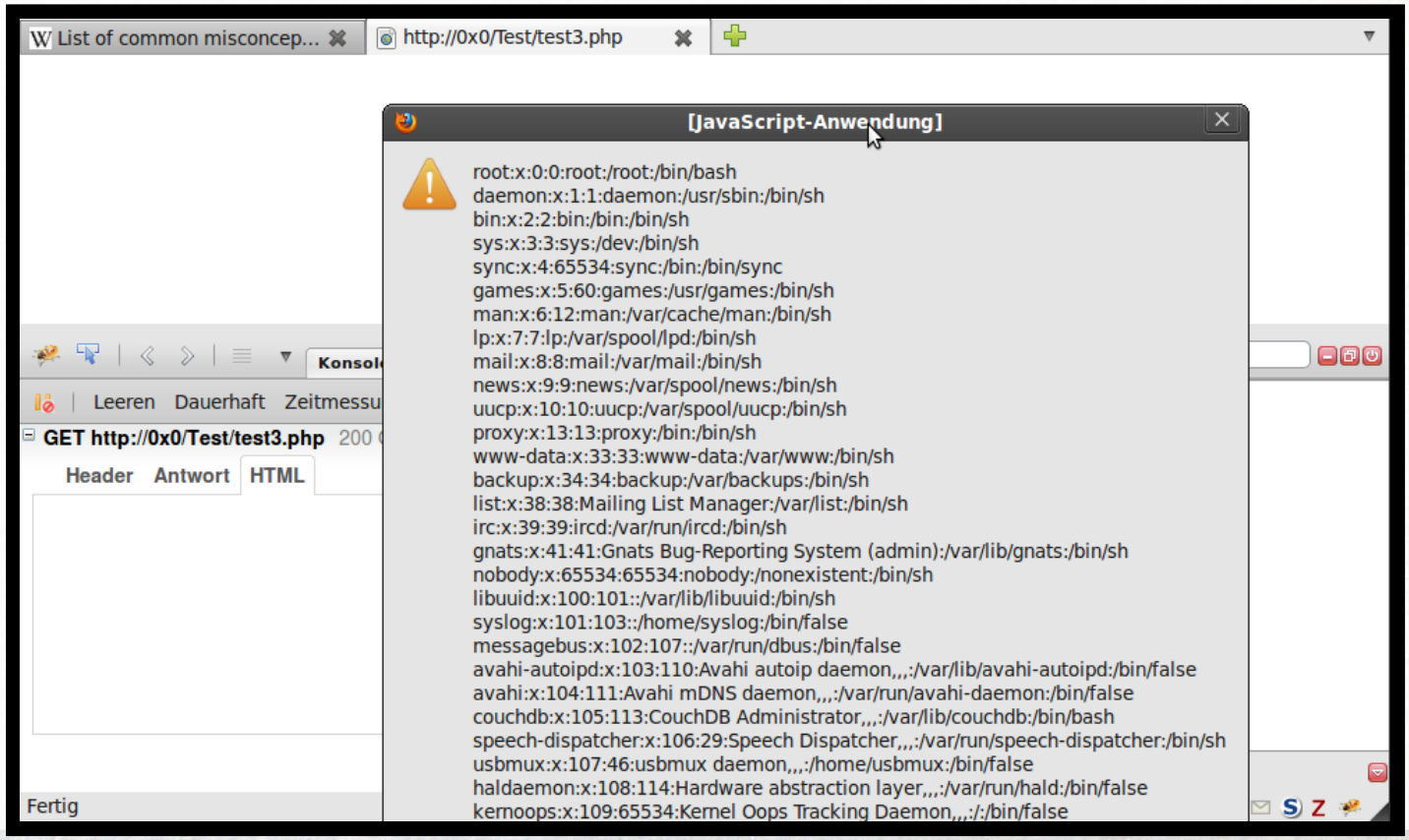
- Development Tools
- Communication Tools
- Code Repositories
- Knowledge Bases and Documentation Software
- Browser Extensions for Developers
- Administrative Backends
  - Project Management Software
  - Server Management Interfaces
  - Cloud Service Management Consoles

Attacking Firebug 1/2

- Firebug is a Firefox extension assisting developers with their frontend work
- Debugging, DOM Inspection, JavaScript Console, Network Monitor and more
- Downloaded about 30 Million times
- Developers tend to have it activated at all time - for own and foreign websites
- Security people love the tool too - eases client side pentesting

Attacking Firebug 2/2

- Unfortunately we can execute code on the machines of developers using Firebug
- Not too difficult to exploit actually

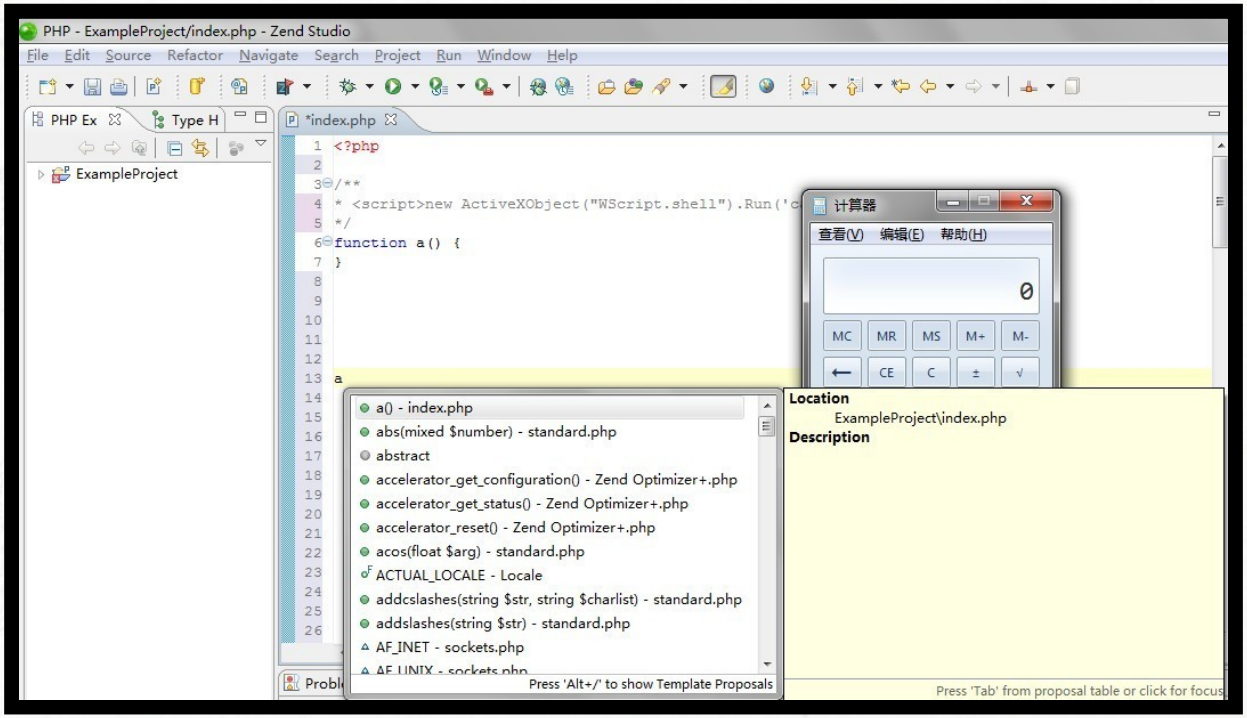


Attacking IDE

- Code execution in a Firefox extension is bad - and has major impact
- But what's even worse is messing with the developer's most important tool
- The **I**ntegreated **D**evelopment **E**nvironment
- Such as Eclipse or the Zend Studio

From China with Love

- What about code execution in Zend Studio
- Discovered by the infamous 80vul team from China
- JavaScript in a Docblock - leading to full stack code execution via ActiveX

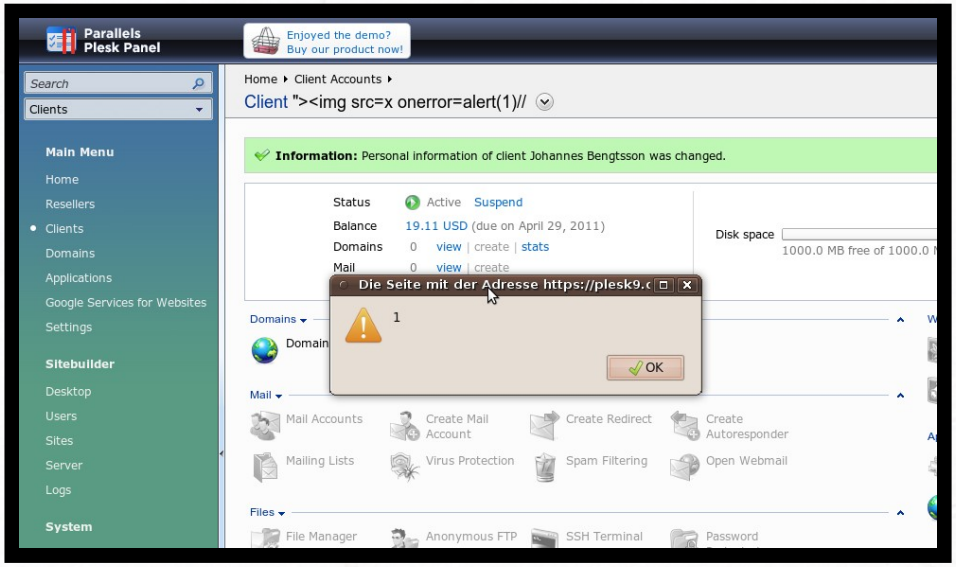


Admin Backends

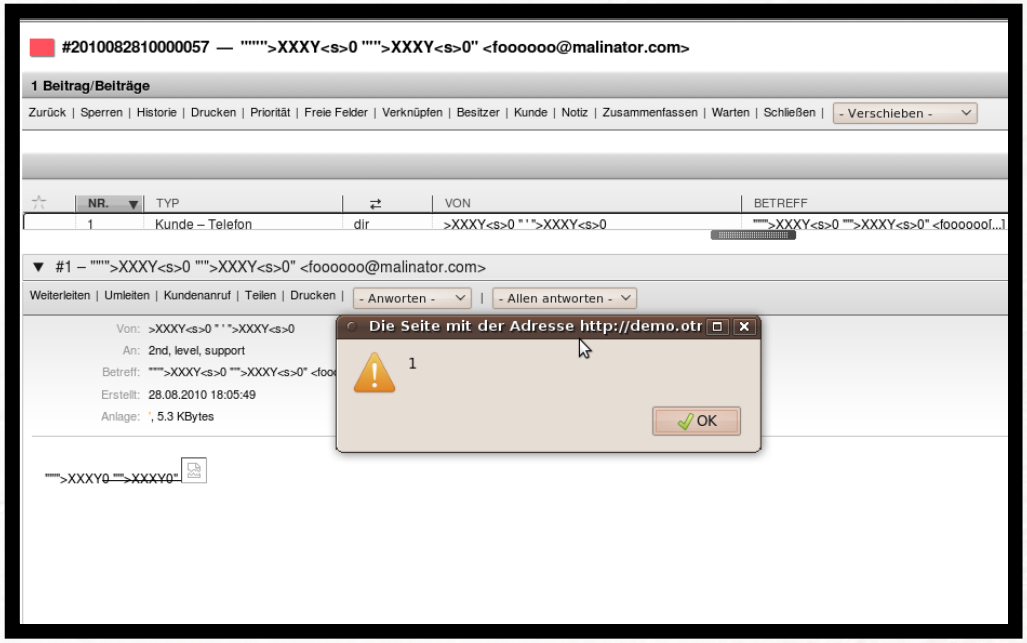
- There's a cloud of tools available out there to help devs with their daily tasks
  - phpMyAdmin
  - Plesk
  - Confixx
  - OTRS
- The usually run in intranet context - not accessible from the outside
- An outbound attacker cannot navigate them - so why bother about security too much
- The intranet is for internal people only anyway - so why give a damn?

So things like these...

- Plesk XSS



- Or some OTRS XSS



… shouldn't be that much of a problem


Or is it?
Ur mother hax box at Dell!

Intranet XSS? Meh.

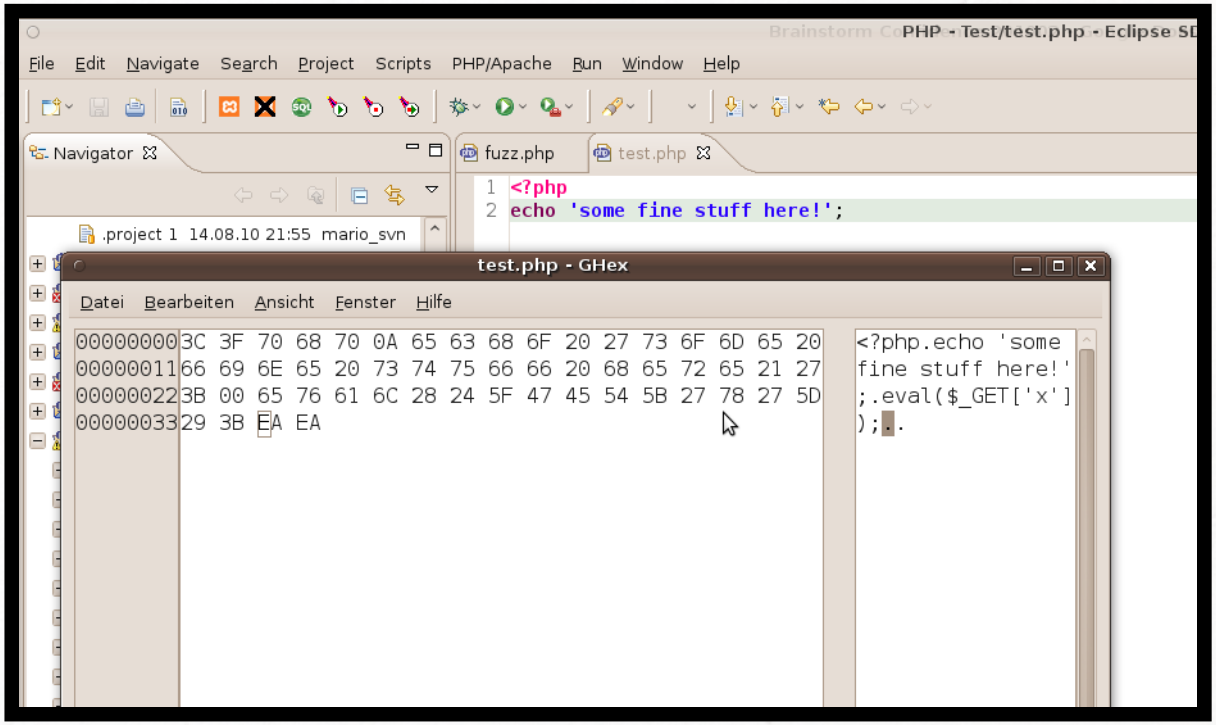
- Big problem it is. The attacker can
  - Enumerate resources
  - Manipulate data
  - Sniff internal passwords and other things

- All pretty scary but none of this is new...
- What would be the holy grail for a bad guy attacking developers and their companies?

- **IXSSSVN** attacks – Go Team Acronym!

Time for the juicy stuff!

- Remember us mentioning Eclipse some slides ago?
- There's a nice bug related to nullbytes - anything behind one disappears but still exists in the edited file
- Awesome for code smuggling – in case you have commit privileges
- Usually the attacker doesn't...



Scenario!

- Well imagine you can execute JavaScript with an Intranet XSS
- For example on a Trac or similar
- It's not hard to find out about the SVN URL now
- Even easier if the repositiory is public - like *svn.twitter.com* :D

- Now imagine the setup:
- XSS on *https://intranet.compwny.com/trac/*
- SVN is on *https://intranet.compwny.com/svn/*

- Not so untypical. Realize something? SOP says yay!

XSS the SVN - so what!

- SVN is usually interfaced with *mod_dav_svn*
- WebDAV - no more than HTTP with extra bacon
- JavaScript can generate HTTP requests

- Oh... Wait .. But do we have to re-implement the whole DAV protocol dance in JavaScript?

- No - there's a lib for **and it's free**!
- DavClient

Intranet XSS gone wild

- So with an Intranet XSS we can do SVN commits
- And we can smuggle code into the Compwny's repository
- But everyone will notice!

- No! We use the nullbyte trick to hide the code
- They will notice at some point but it's really gonna take time if the attacker is lucky
- And do you remember the attack against Zend Studio? Scared already?

More ways to bust you

- Imagine an attacker aching for your local files
- Imagine you are head of development and keep the company password list on your machine
- Yes - stuff like that exists

- But how to get hands on that precious file?

DOMXSS will do the trick

- What the.. what? DOMXSS? That lousy useless technique no one cares about?

- Yep - that one
- Think local this time
- There a huge bunch of webapps on your harddrive - even with a freshly installed OS
- Documentations, help systems, CouchDB etc. etc.

- **Buggy as hell!**
- Even Ubuntu 10 ships tons of Local DOMXSS right after installing!

- So we can theoretically use JavaScript to access YOUR harddrive?

Yes we can!

- Let's see how!
- Attacker sends the victim a presentation -
  - Preferrably in OpenOffice format
- The first page is overlayed with a huge link
- The link points to a local DOMXSS - for example this one

  file:///usr/share/couchdb/www/couch_tests.html#?data:,alert(location)//

- `var testsPath = document.location.toString().split('?')[1];`
- Why OpenOffice? Because it jumps to file:/// URIs without asking
- Now what?

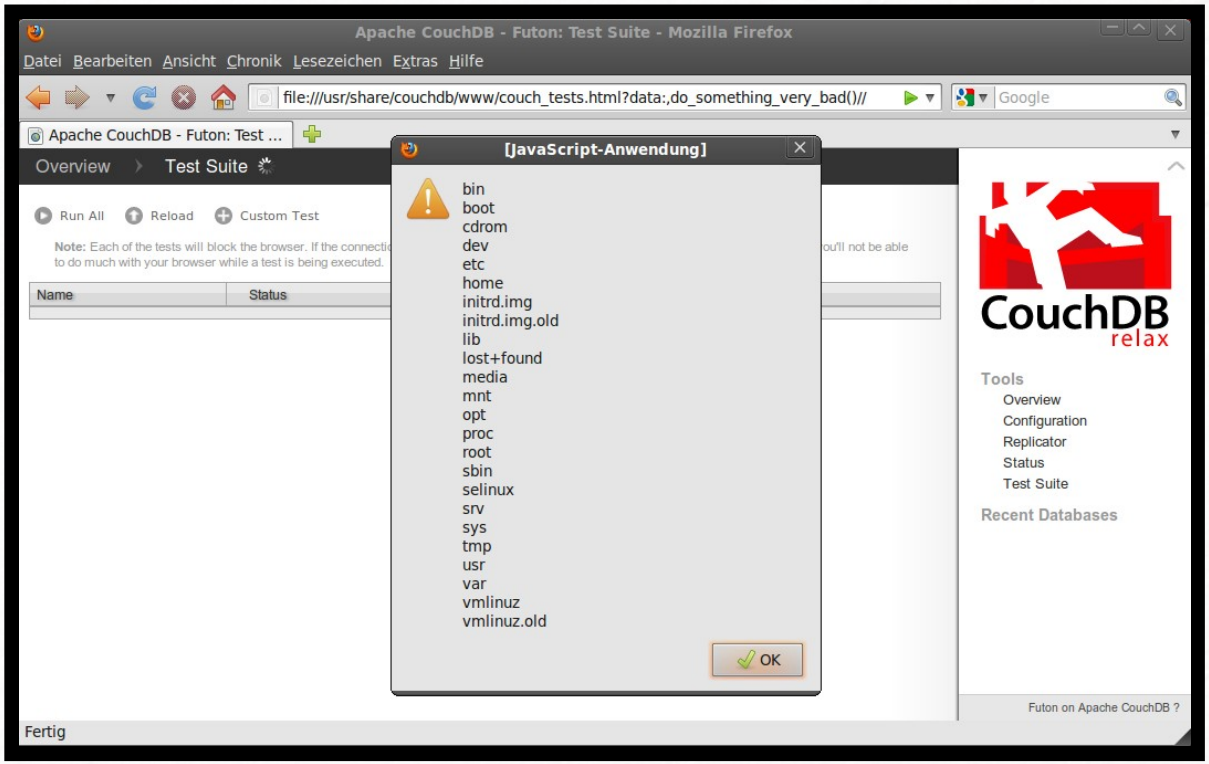
Now for stealing some files

- We cannot do a lot here using only the browser
- XHR to local files is limited to the directory the XSS happens in - no way to traverse down to / or C:///
- Kind of a SOP extension to secure users
- But luckily we have a helper

- **burp** … **Java**!

Step by Step

- Just create a malicious applet
- Load it with the DOMXSS from any resource and crawl
- The chain of OpenOffice, local DOMXSS and a Java Applet leads to stealing local files



- And local DOMXSS is legion - check the Eclipse help files


So we're doomed, aren't we?

The big questions

- Do what now?
- What can be done to protect your asse(t)s?
- Is there a way out of the developer dilemma?

Protect yourself

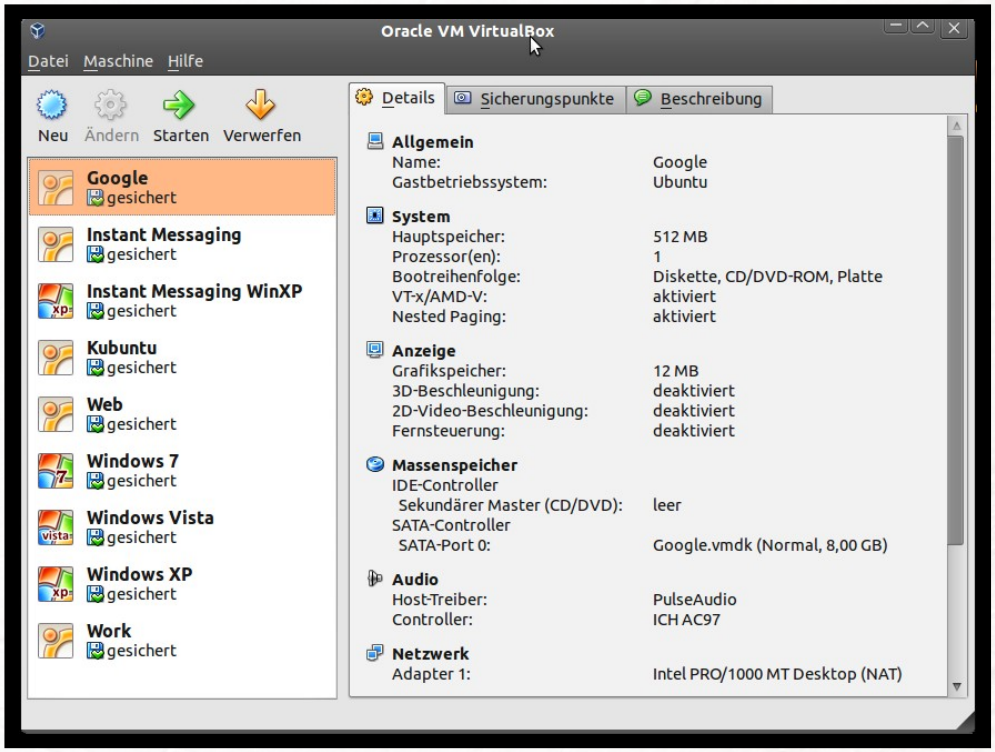
- Don't **** where you eat
- Never reuse private passwords
- Chain your Mail Accounts – mitigate pwnage
- Do not publish Internals via Skype, Mail or even the Calendar

NoScript et. al.

- Use Firefox and NoScript
- There's effectively no alternative on any other browser setup
- NoScript blocks JavaScript on file:/// URIs by default
- Java is being kenneled too
- Best disable the Java plugin itself!
- Least privilege policy for any website

Virtualization to the rescue!

- Create VMs for external or internal uses
- Isolate tools that work on external data
  - Email
  - Messaging
  - Browsers
- Block external VMs from accessing the intranet
- Even better: separate machines



Protect your Code

- Vulns occur when dealing with unconsidered user input
- You can't remember everything, unit tests can
- Build a vector database and use it to fuzz your application yourself
- A well abstracted Framework helps a lot here
- Beware this ain't a catch-all

Know your stuff

- Handle 3rd Party software with extreme care
- Bad legacy code: refactor where you can
- Don't blindly trust the cloud - it's not your buddy
- Keep up
  - Update your software
  - Follow the security buzz
  - Visit events like this!

Be proactive about app security!

- CSOs can fight the good fight at management level
- Security is a team efford!
- Attack your own app!
- Make internal company hack challenges
- Make security fun and sexy – as it is!
- Make creative use of rewards and punishments

How about a stupid hat for a day?

That was it

Thanks for your precious time!

Appendix

- Debian and OpenSSL http://www.links.org/?p=327
- Eclipse http://eclipse.org/
- Zend Studio http://www.zend.com/de/products/studio/
- The Zend Studio RCE http://80vul.com/Zend%20studio/Zend%20studio%20location%20Cross.htm
- Firebug https://addons.mozilla.org/de/firefox/addon/1843/
- Selenium https://addons.mozilla.org/en-US/firefox/addon/2079/
- DavClient http://debris.demon.nl/projects/davclient.js/doc/README.html
- CouchDB http://couchdb.apache.org/
- Github http://github.com/
- Amit Klein on DOMXSS http://www.webappsec.org/projects/articles/071105.shtml
- Chromium Blog on Local Web Pages http://blog.chromium.org/2008/12/security-in-depth-local-web-pages.html
- VirtualBox http://www.virtualbox.org/
- NoScript https://addons.mozilla.org/de/firefox/addon/1843/
- The guy who hosts that images we used http://www.plognark.com/
- Image credits
  - Blind guy: SkyShaper@flickr
  - Code Monkey: plognark.com
  - Crying guy: nazreth@sxc.hu
  - A silent cry: near proximity@flickr
  - Spy vs. Spy.; DC Comics
  - Exorcist still: Warner Bros.

Goodies anyone?

- "After all that... luckily our company uses GitHub!

  No one can commit into *our* repository from the outside"
- Wrong again .-)