PHP Sicherheit
für Administratoren

3. Erlanger Linuxtage
15.01.2005

Alexander Meindl / meindlSOFT
am@meindlsoft.com

PHP Sicherheit: Agenda

Inhalt: PHP Sicherheit

•**Agenda**

•**Motivation**

•**Teil 1:**
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•**Teil 2:**
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•**Teil 3:**
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

Aufgaben des Administrators:

- **sichere Arbeitsumgebung schaffen**
  - PHP Installation / Konfiguration
  - Datensicherheit gewährleisten

- **dem Entwickler sicherheitsrelevante Werkzeuge verfügbar machen**
  - Verschlüsselungsmöglichkeiten

- **präventive Maßnamen**
  - Angriffe oder fehlerhafte PHP
  - Scripte auf Serverebene absichern

Motivation

Inhalt: PHP Sicherheit

Referent:
Alexander Meindl
am@meindlsoft.com

- Statische Angriffe
  - Sicherheitslücken der Serversoftware wird ausgenutzt (Exploits)
  - einfache präventive Maßnahmen:
    - aktuelle Software verwenden
    - Bugtraq Mailingliste lesen
- Dynamische Angriffe
  - nur durch Interaktion können Sicherheitslöcher entdeckt werden
  - komplexe, präventive Maßnahmen:
    - durch Serverkonfiguration
    - durch sicherheitsbewusstes Programmieren

PHP Installation I

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
→**Installation**
→Konfiguration
→Benutzerrechte
→Sessions
→Fehlerbehandlung
→Webserver
→Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
→Verschlüsselung
→openSSL
→Datenbanken

•Teil 3:
Präventive Maßnahmen
→mod_security
→Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- Dezidierten Server immer vorziehen!
- Aktuelle Software verwenden: PHP, Apache, MySQL
- Binäre Pakete oder selbst kompilieren
  - Selbst kompilieren
    - nur benötigte Funktionalität
    - Hardened PHP
    - Ressourcen sparender
    - Know-how wird vorausgesetzt
  - Binärer Pakete
    - einfachere Installation und Updatemöglichkeiten

PHP Installation II

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
→**Installation**
→Konfiguration
→Benutzerrechte
→Sessions
→Fehlerbehandlung
→Webserver
→Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
→Verschlüsselung
→openSSL
→Datenbanken

•Teil 3:
Präventive Maßnahmen
→mod_security
→Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- PHP CGI
  - suPHP (suExec)
  - eigene *php.ini* für jeden *vhost*
  - Webserver Benutzer ist aktueller Benutzer
  - Jeder Skriptaufruf startet eigenen Prozess
- PHP Modul
  - mod_php
  - open_basedir, safe_mode
  - Ressourcensparender

php.ini

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔**Konfiguration**
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- häufig zu finden unter */etc/php.ini*
  - register_global = Off
  - allow_url_fopen = Off
  - enable_dl = Off
- SAPI nutzen
  - unterschiedliche Konfigurationsdateien für verschiedene PHP Modi
  - Beispiel Fedora Core:
    - /etc/php.ini (mod_php)
    - /etc/php-cgi.ini (Shellscript)

Benutzerrechte

Inhalt: PHP Sicherheit

•Agenda

•Motivation

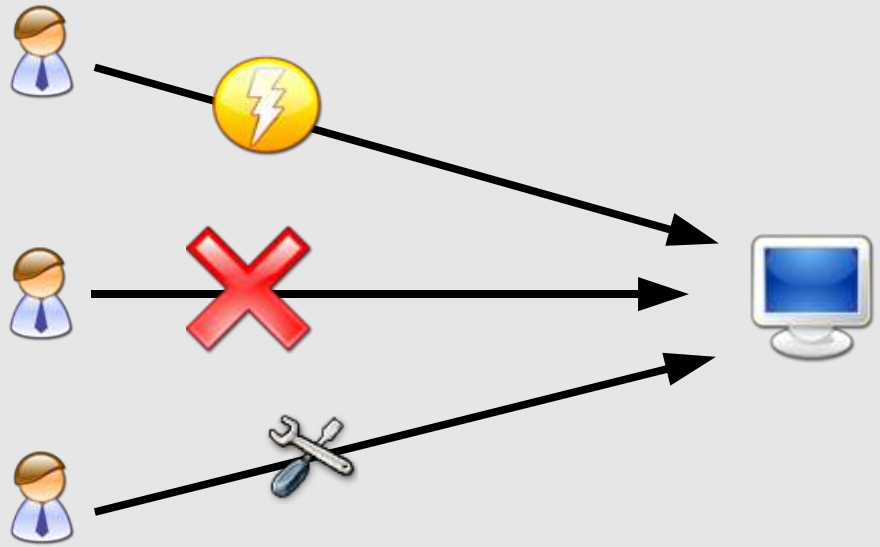
•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔**Benutzerrechte**
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- safe_mode (mögliche Zugriffskonflikte)
- open_basedir = /var/www/htdocs
  - Vorsicht: system('cat /etc/passwd');
- disable_functions = exec,show_source,phpinfo,system,popen, shell_exec



Sessions

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔**Sessions**
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- DB Session vorziehen
- falls dass nicht geht:
  - session.save_path = "/var/cache/php/session"
    - chown root:apache /var/cache/php/session
    - chmod 770 /var/cache/php/session
- session.entropy_file = /dev/urandom
- session.hash_function = 1 (sha1)
- session.use_cookies = 1

Fehlerbehandlung

Inhalt: PHP Sicherheit

•Agenda

•Motivation

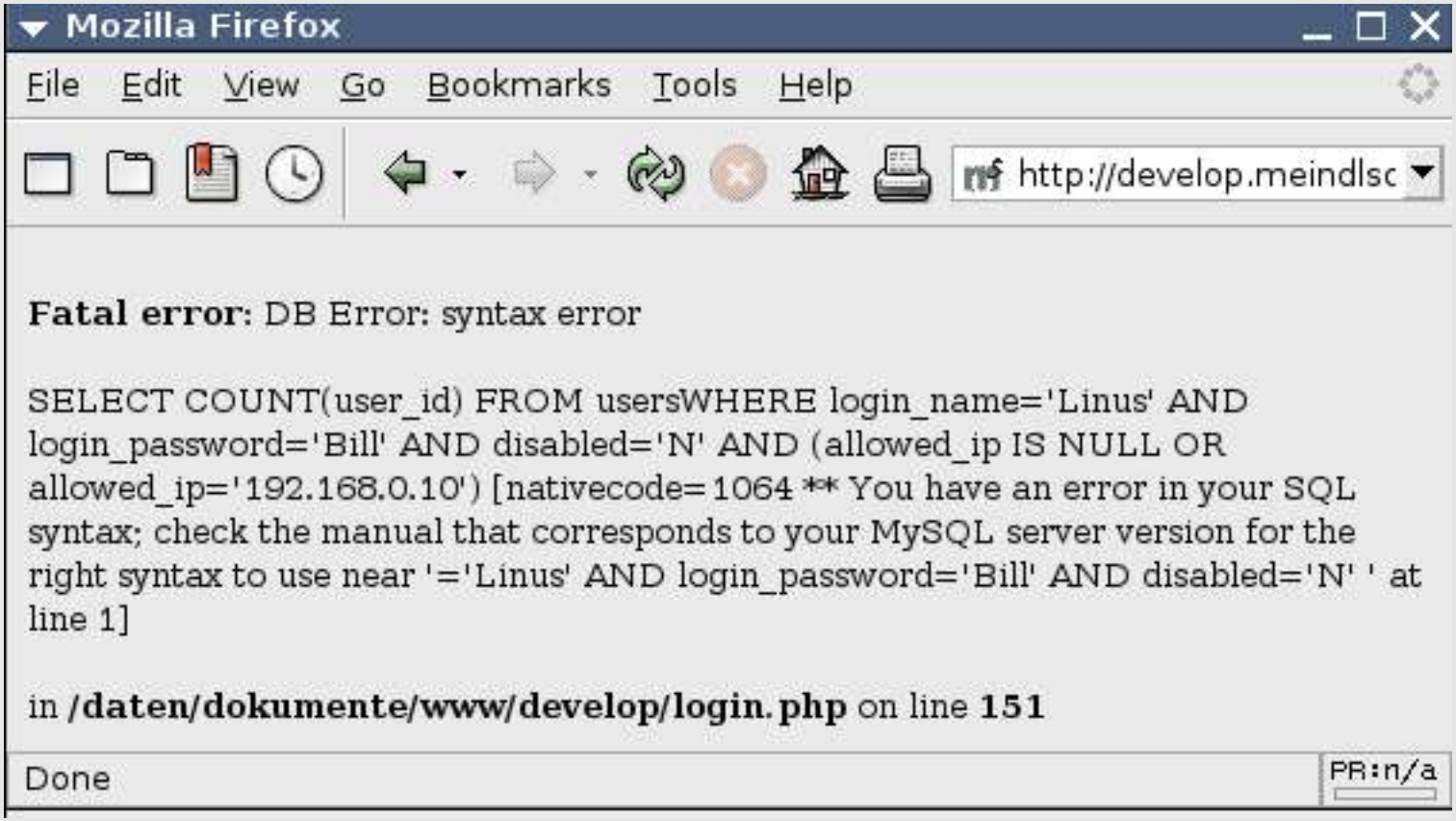
•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔**Fehlerbehandlung**
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- error_reporting = E_ALL
- display_errors = off
- log_errors = on
- error_log="/var/log/php_errors.log"



Webserver Konfiguration

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔**Webserver**
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- php_admin_value
  - open_basedir
  - upload_tmp_dir

- Einstellungen der Dateitypen
  - .php, .phtml usw. als PHP Scripte ausführen
  - PHP Skripte durch Endung „verstecken“ (Security through obscurity)
    - AddType application/x-httpd-php .dhtml
    - expose_php = Off
  - .inc, .class, .conf usw. ausführen verbieten
    - Datenbankkennwörter: config.inc
    - sensible Daten (z.B. Bilder, Dokumente)

Datenbank

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔**Datenbank**

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- Datenbank absichern
  - Netzwerk deaktivieren
    - Socket nutzen
      - Performanter im Vergleich TCP/IP Verb.
      - sicherer, da komplette Netzwerkfunktionaliät nicht genutzt wird
      - muss bei chroot berücksichtigt werden
  - oder nur localhost zulassen
  - oder nur SSL Verbindungen zulassen
  - Anzahl der Verbindungen einschränken

Verschlüsselung

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

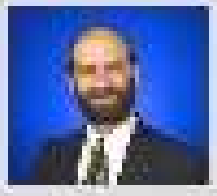
•Teil 2:
Sicherheitsrelevante Werkzeuge
➔**Verschlüsselung**
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- mcrypt
  - Verschlüsselungsalgorithmen wie rijndael (AES), Serpent, blowfish, twofish
- mhash (Einwegverschlüsselung)
  - viele hash Algorithmen
  - HMAC (Hashed Message Authentication Code)
- gpg (Zugriff über system())

kein Anlass für Eigenentwicklungen von eigenen Verschlüsselungsalgorithmen



Jedes Datenelement bleibt so, wie es zuletzt vom authorisierten Benutzer verlassen wurde.

openssl

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔**openSSL**
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- Verschlüsselte Verbindung zum Webserver (https)
- ermöglicht in verschienden PHP Modulen eine verschlüsselte Verbindung
  - curl
  - imap
  - Datenbanken (z.B. pgsql)

Datenbank

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔**Datenbank**

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- sichere Verbindung (z.B. mit SSL)
- DB Transaktionen (innodb)
- DB Verschlüsselung
  - z.B. Oracle unterstützt DB Verschlüsselung
  - wird leider noch nicht von MySQL und PostgreSQL unterstützt
- aktuelle Schnittstellen (wie mysqli oder dbx)

modsecurity

Inhalt: PHP Sicherheit

•Agenda

•Motivation

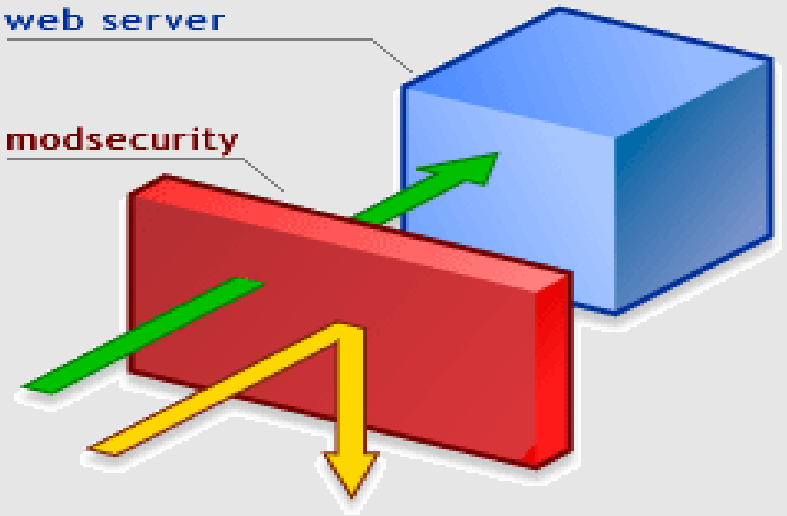
•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔**modsecurity**
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- Präventivmaßnahme gegen CXX
- Präventivmaßnahme gegen SQL Injections
- nicht von Programmierung abhängig!
- über .htaccess steuerbar



modsecurity

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔**mod_security**
➔Monitoring

Referent:
Alexander Meindl
am@meindlsoft.com

- Features
  - Filterung (ähnlich wie reguläre Ausdrücke)
    - z.B. SecFilter "\.bash_history"
  - Logging
  - Chroot Funktion
    - Vorteil gegenüber chroot:
      - weniger Verwaltungsaufwand und HDD Speicher
    - Nachteil:
      - kein Schutz vor Apache Exploits

Monitoring

Inhalt: PHP Sicherheit

•Agenda

•Motivation

•Teil 1:
Sichere Arbeitsumgebung schaffen
➔Installation
➔Konfiguration
➔Benutzerrechte
➔Sessions
➔Fehlerbehandlung
➔Webserver
➔Datenbank

•Teil 2:
Sicherheitsrelevante Werkzeuge
➔Verschlüsselung
➔openSSL
➔Datenbanken

•Teil 3:
Präventive Maßnahmen
➔mod_security
➔**Monitoring**

Referent:
Alexander Meindl
am@meindlsoft.com

- Zend Plattform
  - Fehler Überwachung von Webserver / Datenbank / PHP
  - Performance Überwachung Datenbank / Webserver / PHP
  - Caching und Komprimierung
  - und bietet viele Features für Entwickler

Top Alerts

Id	Description	Last Occurence	Server	Vhost	File Name	Severity
2	Slow Query Execution	24 Oct 2004, 13:50:59	10.10.10.10	vhost1	sql.php	Severe
1	Load Average	24 Oct 2004, 10:19:32	10.10.10.10			Severe
8	PHP Error	24 Oct 2004, 13:37:27	localhost	www.zend.com	.../notfound.php	Severe
7	Slow Function Execution	24 Oct 2004, 13:26:10	localhost	www.zend.com	.../add_user.php	Severe
5	Inconsistent Output Size	24 Oct 2004, 13:04:16	localhost	www.zend.com	.../read.php	Severe

[Abbildung: Zend Plattform]

PHP Sicherheit - Danke!

Definition des Begriffs „Sicherheit“:

- „Sicherheit ist kein Zustand, sondern ein Prozess“, von *Wau Holland*
- „Wie eine Kette ist Sicherheit nur so stark wie ihr schwächstes Glied“, von *Bruce Schneier*

Gibt es Fragen?

Für weitere Auskünfte stehe ich Ihnen nach dem Vortrag zur Verfügung!

Weitere Ressourcen und Downloads der Vortragsunterlagen:
http://www.meindlsoft.de/php/