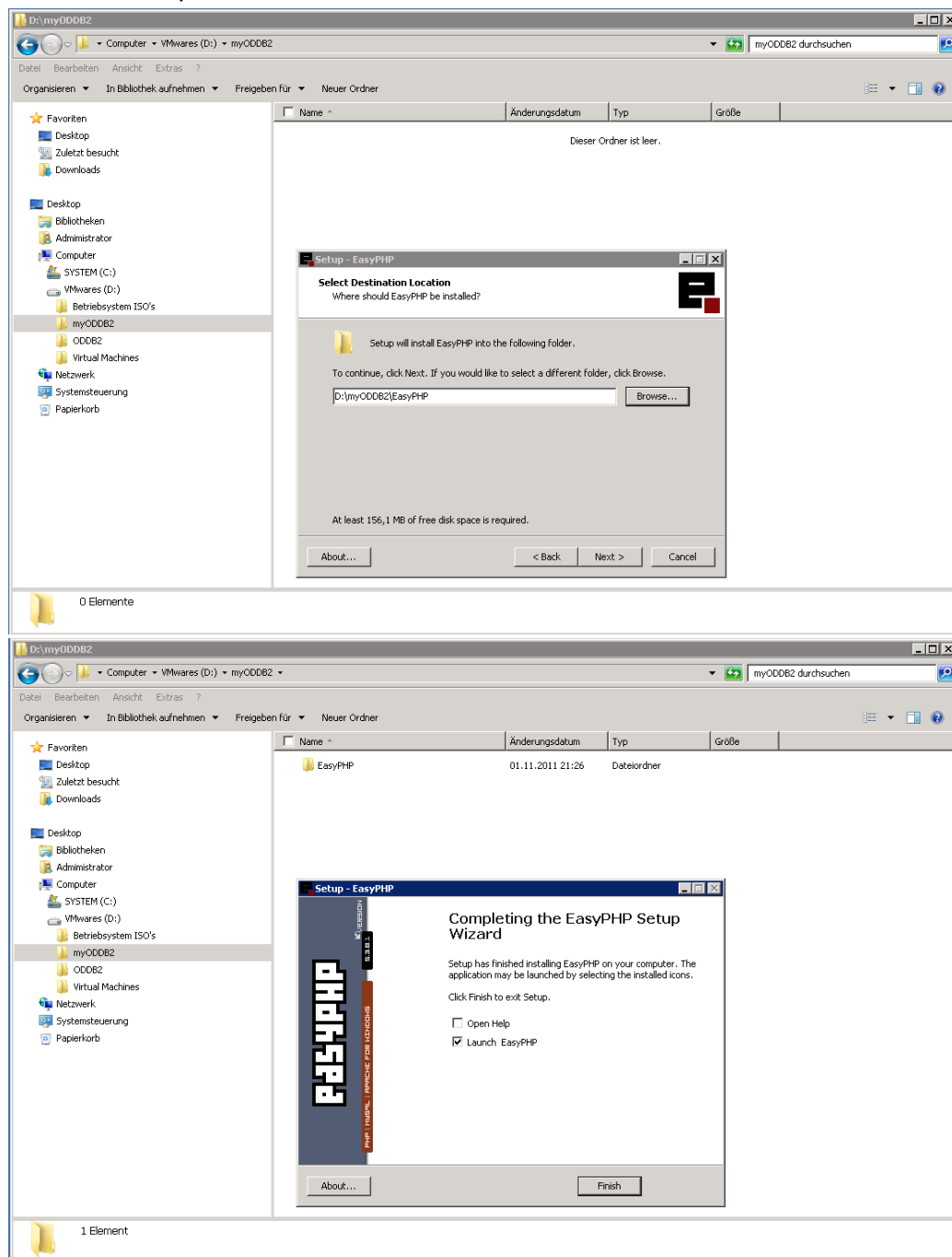


Hier wird die Installation einer ODDb2 auf einem Windows7 oder Windows Server 2008 R2 gezeigt.

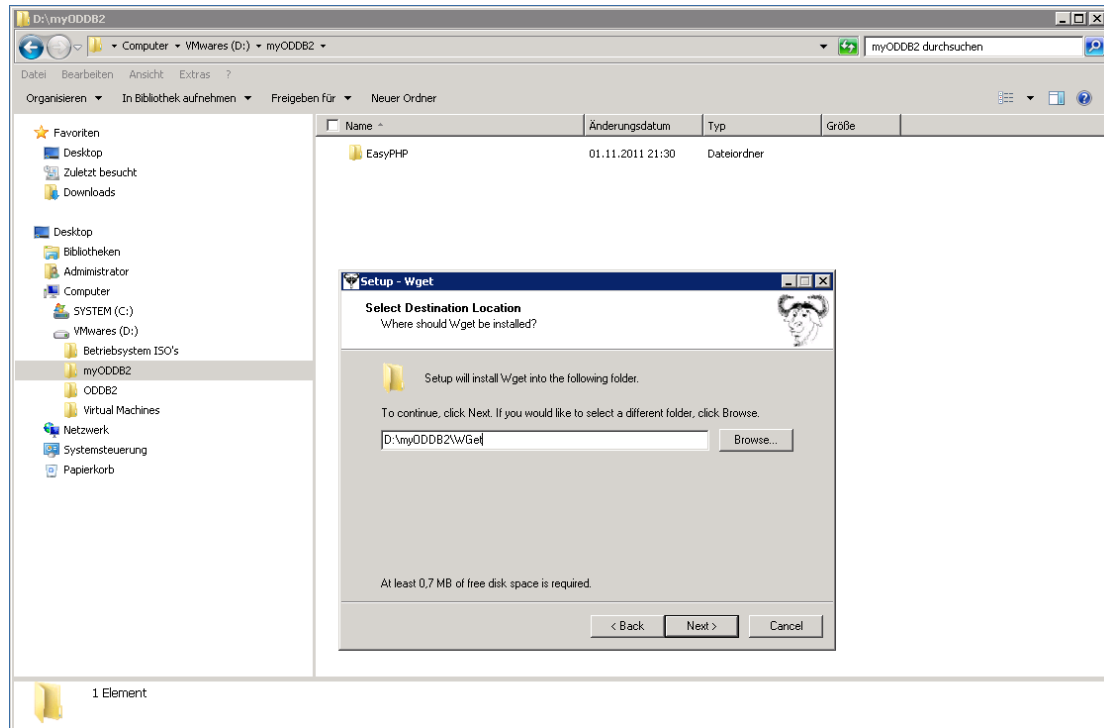
Ich benutze das EasyPHP Paket um die oddb2 auszuführen. EasyPHP ist nicht für „Produktiveinsätze“ gedacht, und auch **nur aktiv wenn ein Benutzer an Windows angemeldet ist (also läuft nicht als Dienst)**. Ein einfacherer Weg um die oddb2 unter Windows ans Laufen zu bringen ist mir aber leider nicht eingefallen

1. Ein Verzeichnis anlegen in das die Bestandteile der ODDb2 hinein installiert werden.  
Dies dient zur besseren Übersicht.  
Natürlich kann auch alles in den Standard Programme Ordner hinein installiert werden.  
Hier in meinem Beispiel wird alles in D:\myODDB2 hinein installiert.

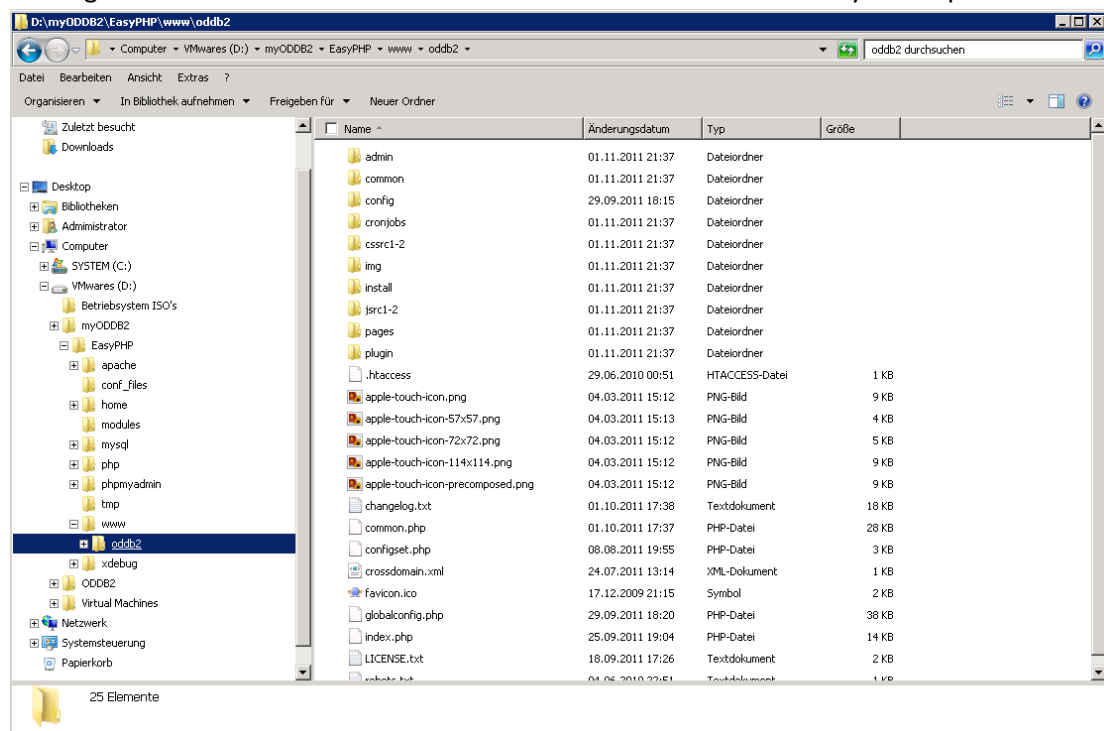
2. Zuerst wird EasyPHP in dieses Verzeichnis hinein installiert



### 3. Nun folgt die Installation von WGet für Windows



### 4. Die eigentlichen oddb2 Dateien in das www Unterverzeichnis von EasyPHP kopieren



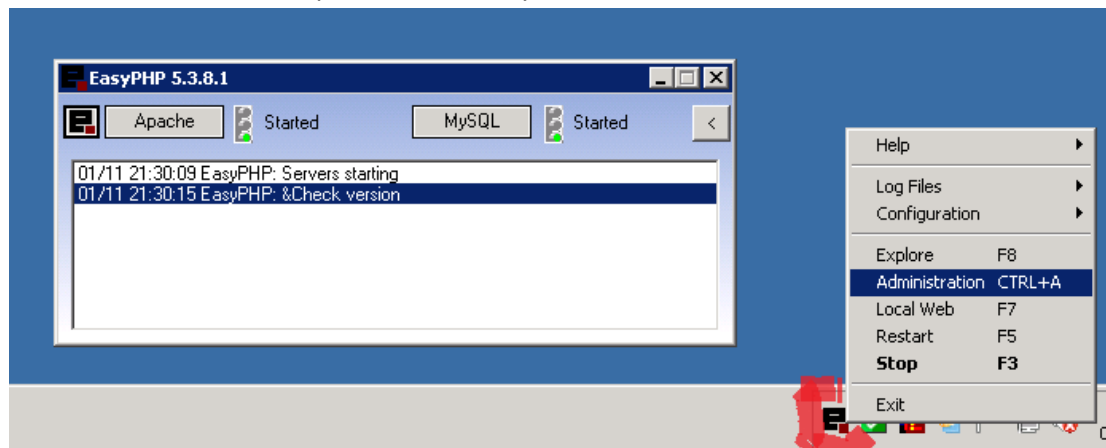
Abhängig davon ob die Dateien mit oder ohne den oddb2 Ordner in den www Ordner eingefügt werden ist die Datenbank auf dem Server direkt unter `www.DeinServer.de/` , oder unter `www.DeinServer.de/oddb2/` zu erreichen.

Bei meiner Beispielininstallation entscheide ich mich dafür alles in einen Unterordner oddb2 zu packen, da mir dies übersichtlicher erscheint.

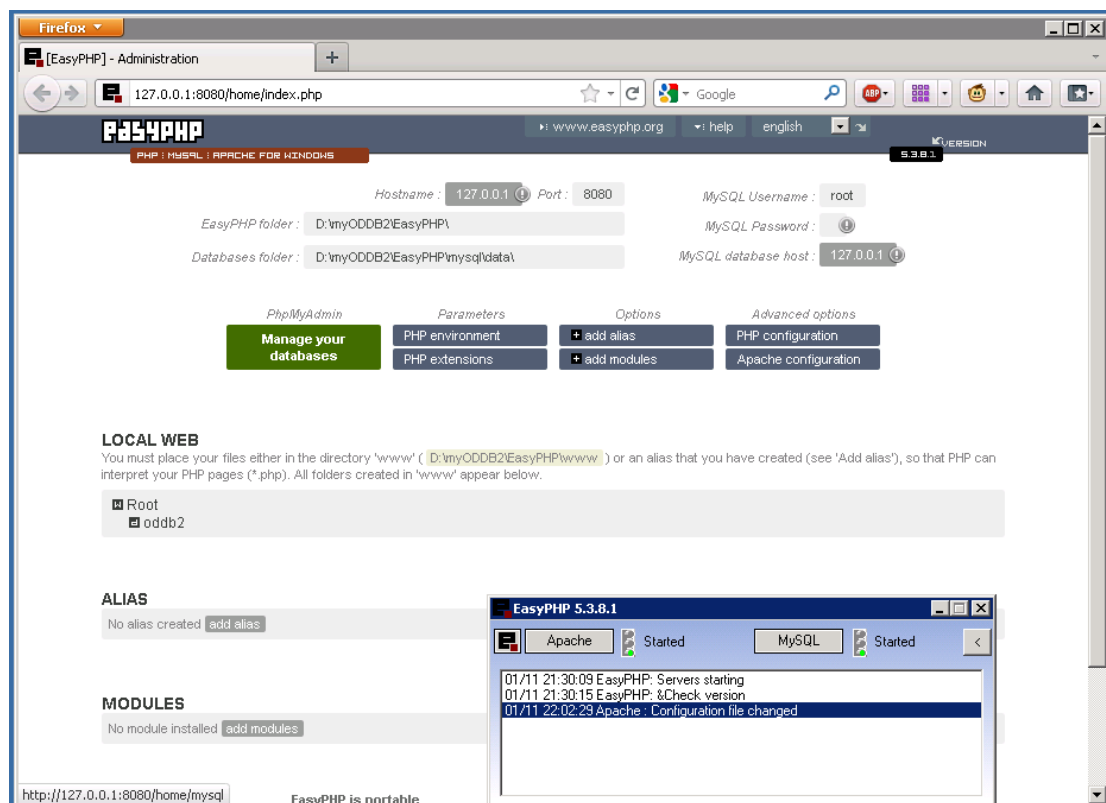
5. Weiter geht es mit der Konfiguration von EasyPHP. Grundsätzlich läuft die oddb2 schon jetzt, aber es sind noch einige wichtige Einstellungen, wie z.B. das Setzen eines Passwortes für den mit EasyPHP mit installierten MySQL Server dringend anzuraten.

EasyPHP hat zwar bei der Installation erkannt das in meinem Beispiel Port 80 schon in Benutzung ist, und sich und Apache auf Port 8080 konfiguriert, aber auch das kann man auf einen anderen Port ändern wenn man denn mag.

Ein Doppelklick auf das TrayIcon von EasyPHP bringt dessen Statusfenster hervor.  
Hier sollten nun sowohl Apache als auch MySQL auf Started stehen

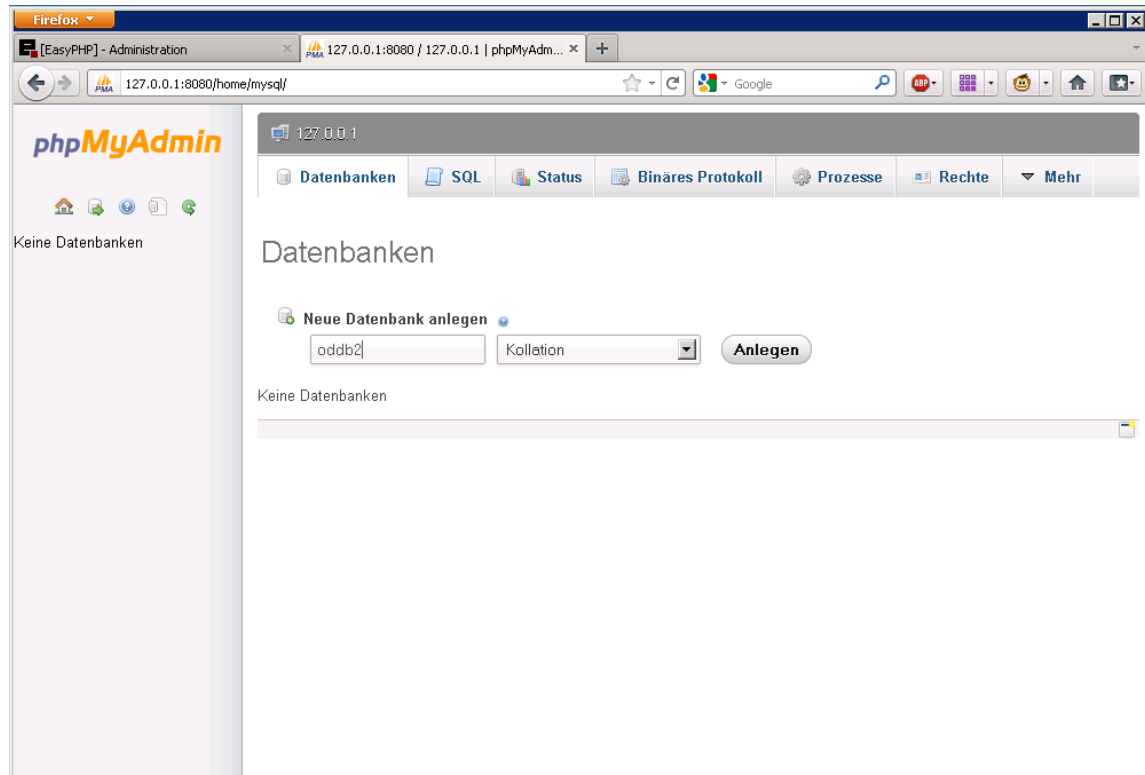


Über das Kontextmenü von EasyPHP ist nun die Administrationsoberfläche im WebBrowser aufzurufen

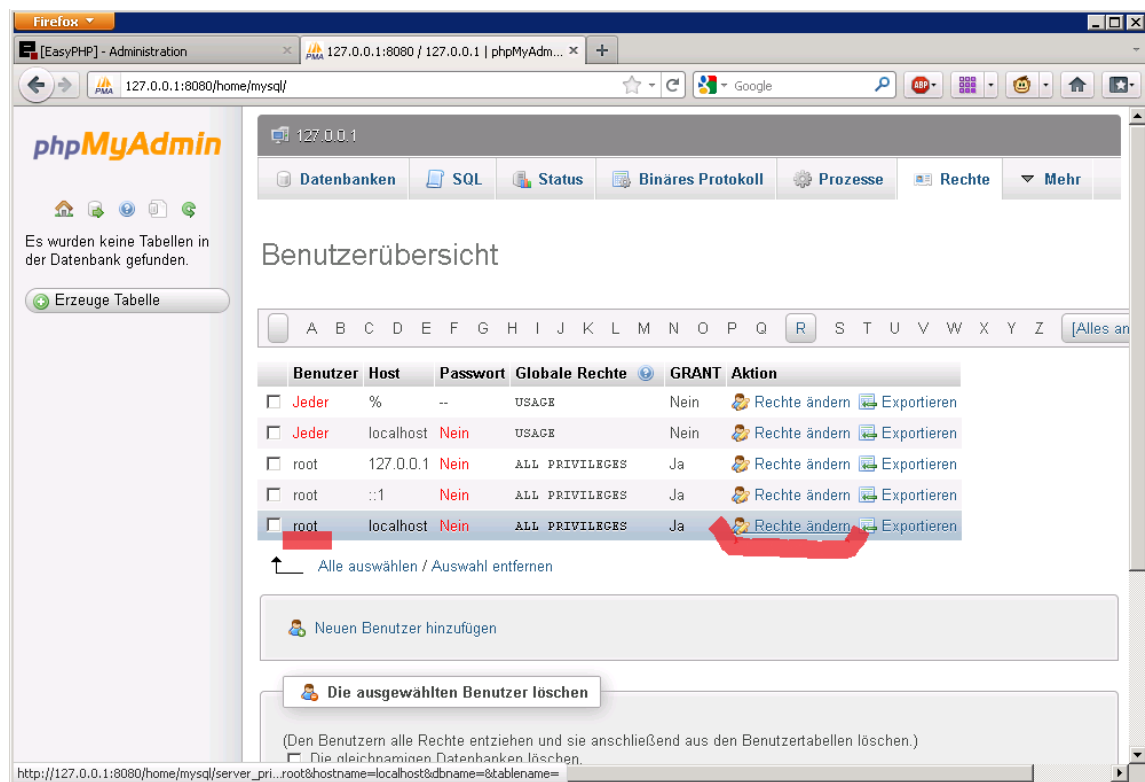


Mit dem grün hinterlegten Manage your databases geht es jetzt weiter mit phpMyAdmin

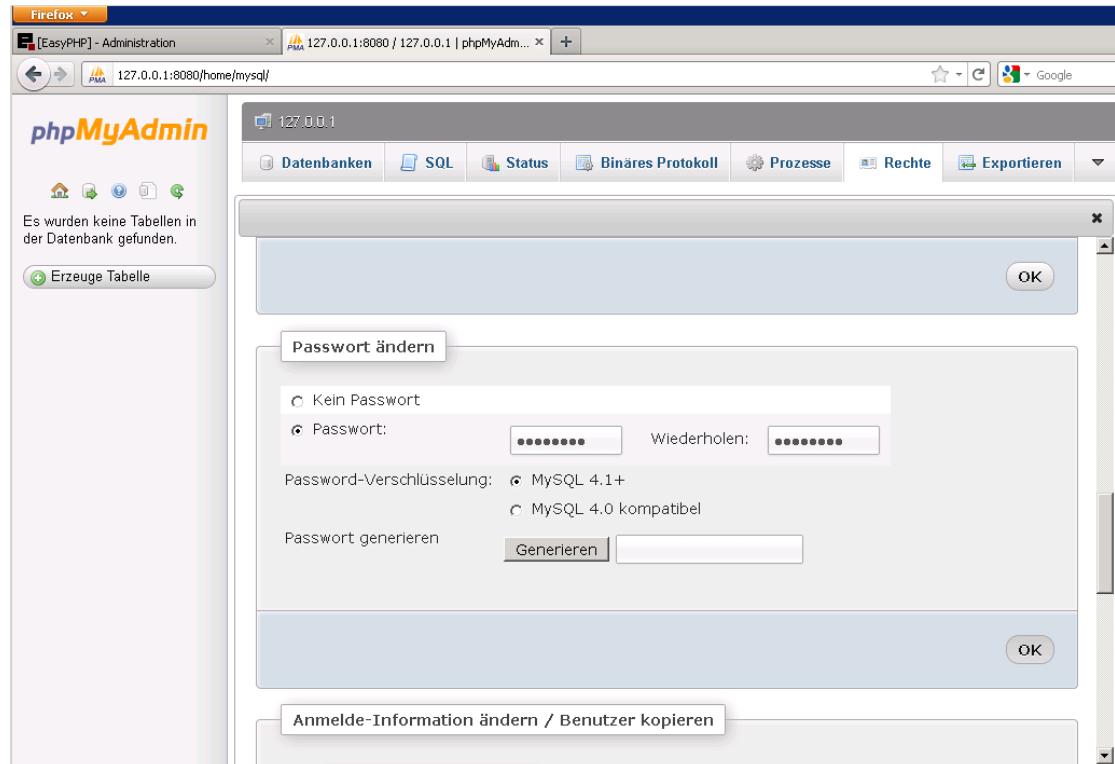
wo in der Reiterkarte Datenbanken nun eine neue Datenbank namens oddb2 angelegt wird



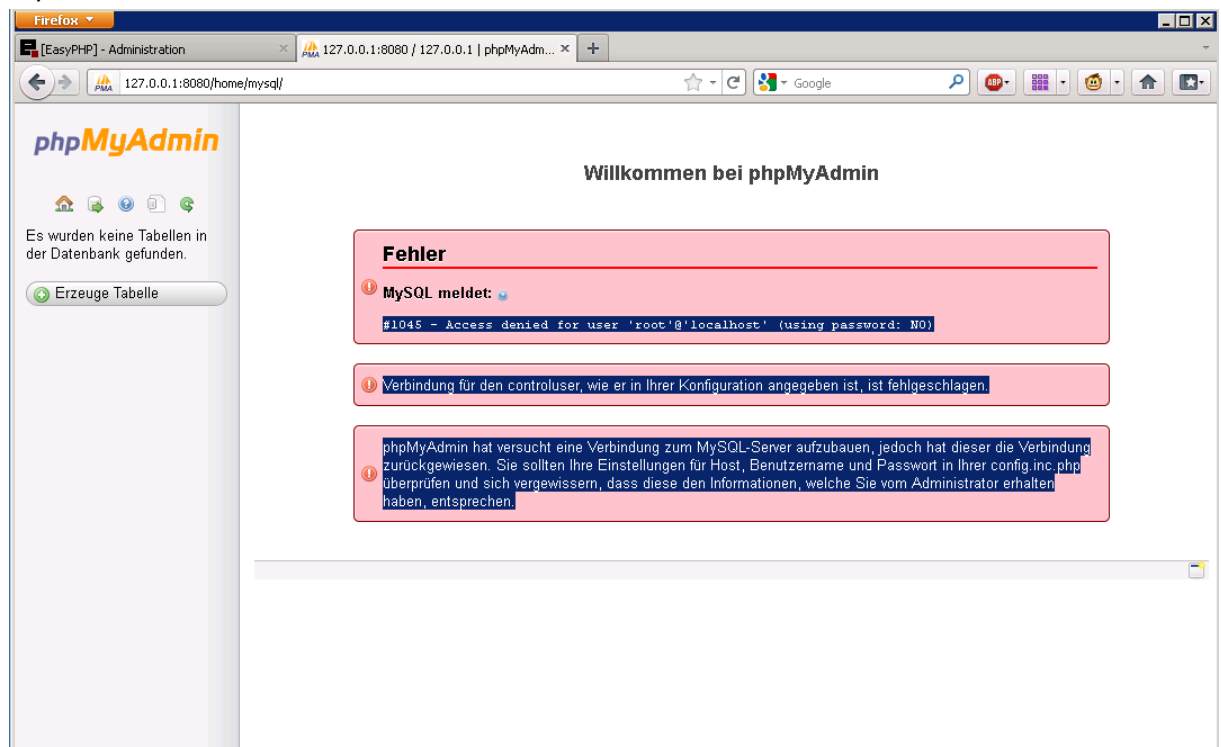
Nun sollte unbedingt noch ein Passwort für den mySQL Administrator root gesetzt werden.  
In der Reiterkarte Rechte in einer der Zeilen des Benutzers root auf Rechte ändern klicken.



Recht weit unten findet sich das Eingabefeld für das Passwort samt zugehörigem OK Button.

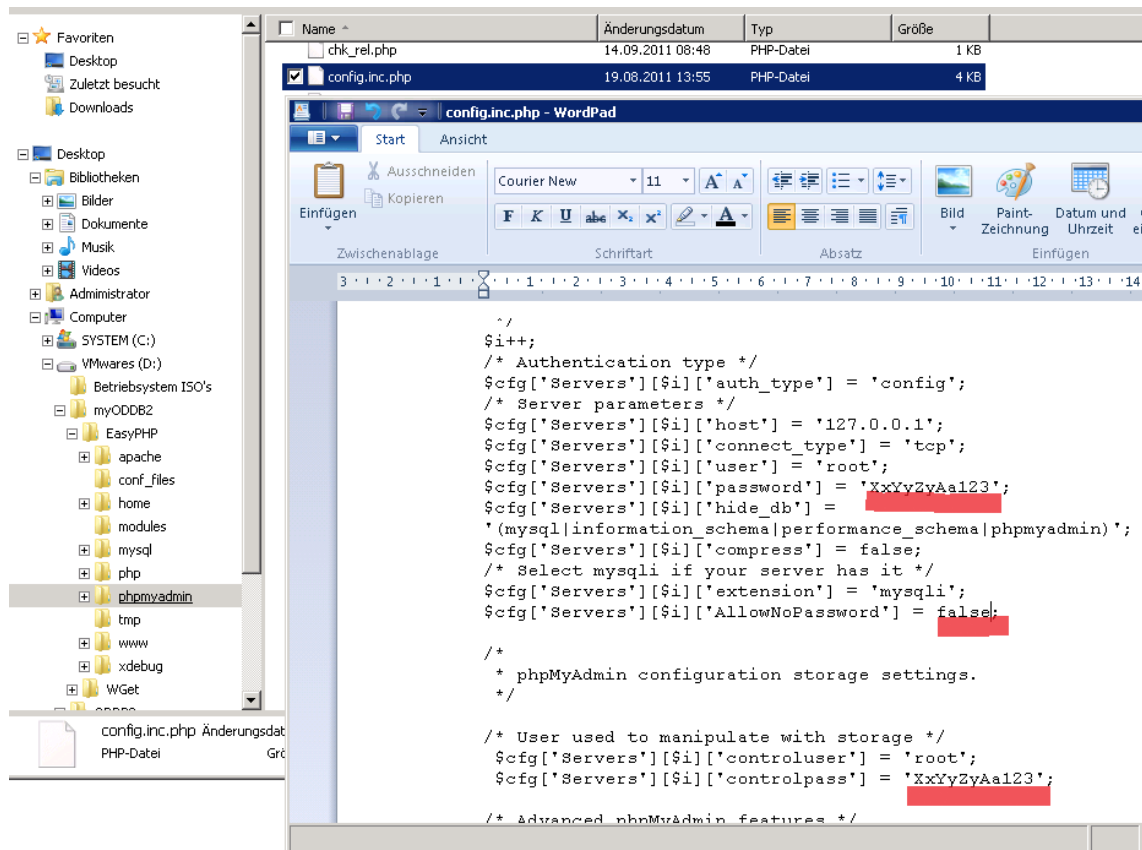


Nun haben wir uns erst mal aus phpMyAdmin ausgesperrt da dies noch KEIN Passwort für den MySQL Administrations Account root verwendet



Zwar nicht zwingend für den Betrieb der oddb2 erforderlich, aber doch im Fall des Falles nice to have noch mal auf Einfachem Wege an die MySQL Konfiguration zu kommen.

In der Datei EasyPHP\phpMyAdmin\config.inc.php wird das Passwort für den mySQL root hinterlegt und ob überhaupt eins verwendet werden soll



Die Datei ist am besten mit Wordpad zu öffnen.

Bei notepad ect. ist das nur sehr unübersichtlicher Zeichensalat.

In den Beiden Zeilen

```
$cfg['Servers'][$i]['password'] =  
$cfg['Servers'][$i]['controlpass'] =
```

Das eben für den root vergebene Passwort eintragen und in der Zeile

```
$cfg['Servers'][$i]['AllowNoPassword'] =
```

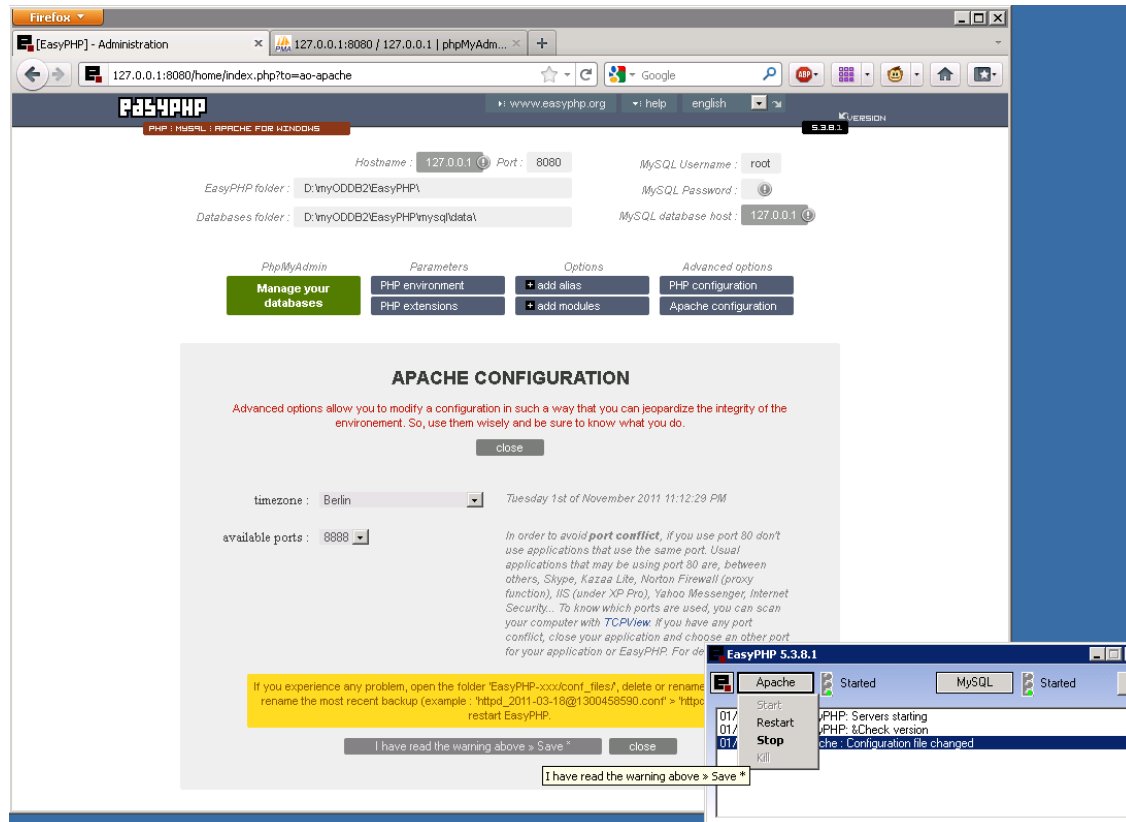
True auf false ändern.

Nach dem Abspeichern der Datei funktioniert auch phpMyAdmin wieder.

Als letztes sollte noch unter Apache Configuration die Zeitzone auf Berlin gestellt werden.

Hier kann man auch einen anderen Port auf dem der Apache auf Anfragen lauscht ändern.

Ich habe hier jetzt Port 8888 gesetzt.



## 6. Die Windows Firewall

Da der Apache Server der Datenbank auch von außen erreichbar sein soll muss noch eine entsprechende Firewall Regel definiert werden.

Ich werde hier nicht detailliert und bebildert darauf eingehen da das den Rahmen sprengen würde.

Unter Windows7 und Server 2008 R2 in der Windows Firewall mit erweiterter Sicherheit

Eingehende Regeln

Neue Regel

Port

TCP / Bestimmte Ports = der der für den Apache konfiguriert wurde

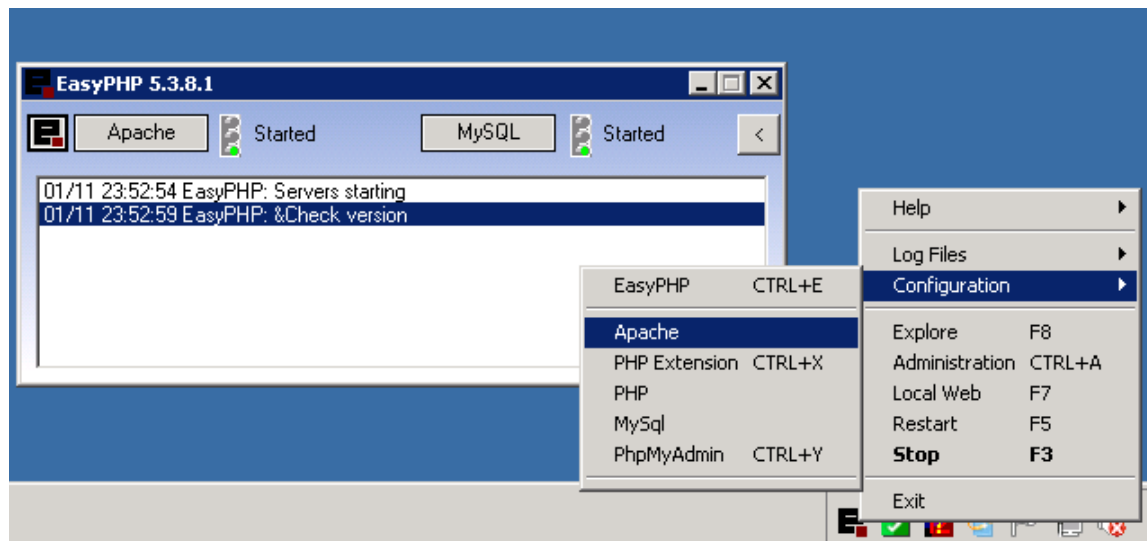
Verbindung zulassen

Anhaken für welche Netzkonfigs die Regel gelten soll

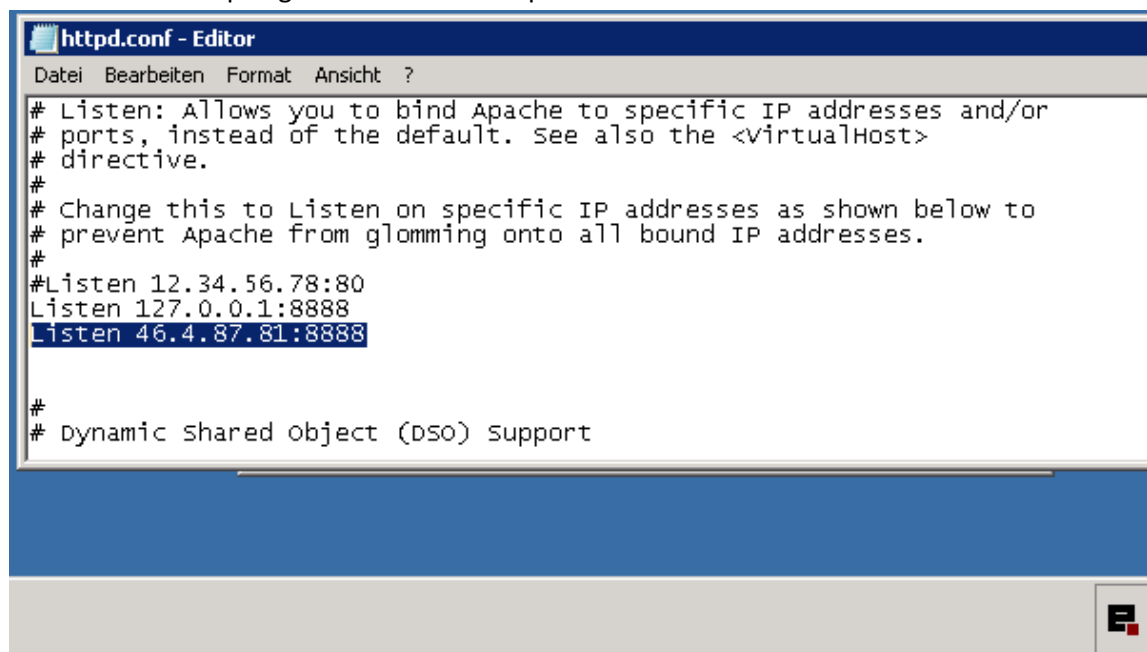
Der Regel einen Namen geben

Dem Apachen muss außerdem gesagt werden das er außer auf die Standardmäßig im EasyPHP Paket deklarierte Localhost IP 127.0.0.1 auch auf die IP Adresse des Richtung Internet führenden Interfaces lauschen soll.

Über das Kontextmenü von EasyPHP ist die Konfigurationsdatei des Apache Web Servers erreichbar



In der nun in Notepad geöffneten Datei httpd.conf



Muss unterhalb der Listen 127.0.0.1:Port des Apache eine weitere Listen Zeile mit der IP Adresse des Richtung Internet führenden Interfaces samt Port auf dem der Apache lauscht eingefügt werden.

Man kann Apache auch an allen Interfaces bzw. deren IP Adresse lauschen lassen, aber das halte ich für keine gute Idee.

Die Entwickler des Apache offensichtlich auch nicht wie man den Kommentarzeilen entnehmen kann.

Im Normalfall bemerkt EasyPHP die Konfigurationsänderung selber und startet den Apache neu, falls nicht über das Statusfenster oder das Kontextmenü von EasyPHP den Restart von Hand auslösen.



## 7. Nun kommt die Grundinstallation der oddb2

Auf diese wird in der offiziellen Installationsanleitung näher eingegangen.

### Installations-Einstellungen

Dieses Script speichert die wichtigsten Grundeinstellungen der ODDb und legt die erste Instanz an.  
Die Einstellungen kannst du später in der globalconfig.php wieder ändern, weitere Instanzen kannst du in der Administrationsoberfläche anlegen.

vollständige Adresse (mit /)	<input type="text" value="http://www.Server:8888/od"/>
Server (ohne /)	<input type="text" value="www.Serverr.de:8888"/>
Sicherheits-Schlüssel	<input type="text"/> <small>(für die Cronjobs: bitte nur Buchstaben und Zahlen)</small>
MySQL-Server	<input type="text" value="localhost"/>
MySQL-Benutzer	<input type="text" value="root"/>
MySQL-Passwort	<input type="password" value="....."/>
MySQL-Datenbank	<input type="text" value="oddb2"/>
MySQL-Tabellenpräfix	<input type="text" value="oddb_"/>
Administrator-UserID	<input type="text"/> <small>(deine OD-Spieler-ID)</small>

Installation starten

8. Das Thema cronjobs bzw. das Ersetzen der Linux Cronjobs durch den Windows Aufgabenplaner. Bugfix hat 2 cronjobs, also Aufgaben die in bestimmten Abständen oder zu einer bestimmten Uhrzeit ausgeführt werden müssen vorgesehen.  
Dies passiert durch Abrufen der cronjobs\1.php und 2.php Dateien

Der Windows 7 und Windows Server 2008 R2 Aufgabenplaner kann alles Mögliche, aber keine Dateien von Web Servern abrufen.  
Dies erledigt das zu Anfang installierte WGet für uns..

Leider kann der Aufgabenplaner kein Intervall von 1 Minute wie Bugfix es für den ersten cronjob vorgesehen hat.

Oder besser gesagt 1 Minute ist nicht auswählbar.  
5 Minuten sind das kleinste Intervall.

Über in .xml Dateien vorkonfigurierte Aufgaben geht es aber.

In den Dateien ist alles bis auf Serveradresse:Port und Programmverzeichnis von WGet.  
In den mit Notepad zu bearbeitenden Dateien, die hier angehängt sind, muss also folgendes angepasst werden

```
<Command>"C:\Programmpfad zu WGet\wget.exe"</Command>
<Arguments>"http://Server:Port/oddb2/cronjobs/1.php?key=DerCronjobKey"</Arguments>
```

Also z.B.

```
<Command>"D:\myODDB2\WGet\bin\wget.exe"</Command>
<Arguments>"http://127.0.0.1:8888/oddb2/cronjobs/1.php?key=DerCronjobKey"</Arguments>
```

Der Wert **DerCronjobKey** wird während der Installation der oddb2 in Schritt 7 vergeben.  
Ohne genau diesen Key laufen die Cronjobs bzw. die Aufgaben im Windows Aufgabenplaner nicht.

Hier eine Beispiel-Datei (Ausführung jede Minute):

```
<?xml version="1.0" encoding="UTF-16"?>
<Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
  <RegistrationInfo>
    <Date>2011-11-01T19:00:43.4942306</Date>
    <Author>tmm</Author>
  </RegistrationInfo>
  <Triggers>
    <CalendarTrigger>
      <Repetition>
        <Interval>PT1M</Interval>
        <Duration>P1D</Duration>
        <StopAtDurationEnd>false</StopAtDurationEnd>
      </Repetition>
      <StartBoundary>2011-01-01T05:00:00</StartBoundary>
      <Enabled>true</Enabled>
      <ScheduleByDay>
```

```

        <DaysInterval>1</DaysInterval>
    </ScheduleByDay>
</CalendarTrigger>
</Triggers>
<Principals>
    <Principal id="Author">
        <UserId>S-1-5-18</UserId>
        <RunLevel>HighestAvailable</RunLevel>
    </Principal>
</Principals>
<Settings>
    <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
    <DisallowStartIfOnBatteries>false</DisallowStartIfOnBatteries>
    <StopIfGoingOnBatteries>false</StopIfGoingOnBatteries>
    <AllowHardTerminate>true</AllowHardTerminate>
    <StartWhenAvailable>false</StartWhenAvailable>
    <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable>
    <IdleSettings>
        <StopOnIdleEnd>true</StopOnIdleEnd>
        <RestartOnIdle>false</RestartOnIdle>
    </IdleSettings>
    <AllowStartOnDemand>true</AllowStartOnDemand>
    <Enabled>true</Enabled>
    <Hidden>false</Hidden>
    <RunOnlyIfIdle>false</RunOnlyIfIdle>
    <WakeToRun>false</WakeToRun>
    <ExecutionTimeLimit>PT1H</ExecutionTimeLimit>
    <Priority>7</Priority>
</Settings>
<Actions Context="Author">
    <Exec>
        <Command>"C:\Programmpfad zu WGet\wget.exe"</Command>

<Arguments>"http://Server:Port/oddb2/cronjobs/1.php?key=DerCronjobKey"</Arguments>
    </Exec>
</Actions>
</Task>

```

Für den zweiten Cronjob (1x am Tag) den <Triggers>-Bereich durch Folgendes ersetzen:

```

<Triggers>
    <CalendarTrigger>
        <StartBoundary>2011-01-01T04:00:00</StartBoundary>
        <Enabled>true</Enabled>
        <ScheduleByDay>
            <DaysInterval>1</DaysInterval>
        </ScheduleByDay>
    </CalendarTrigger>
</Triggers>

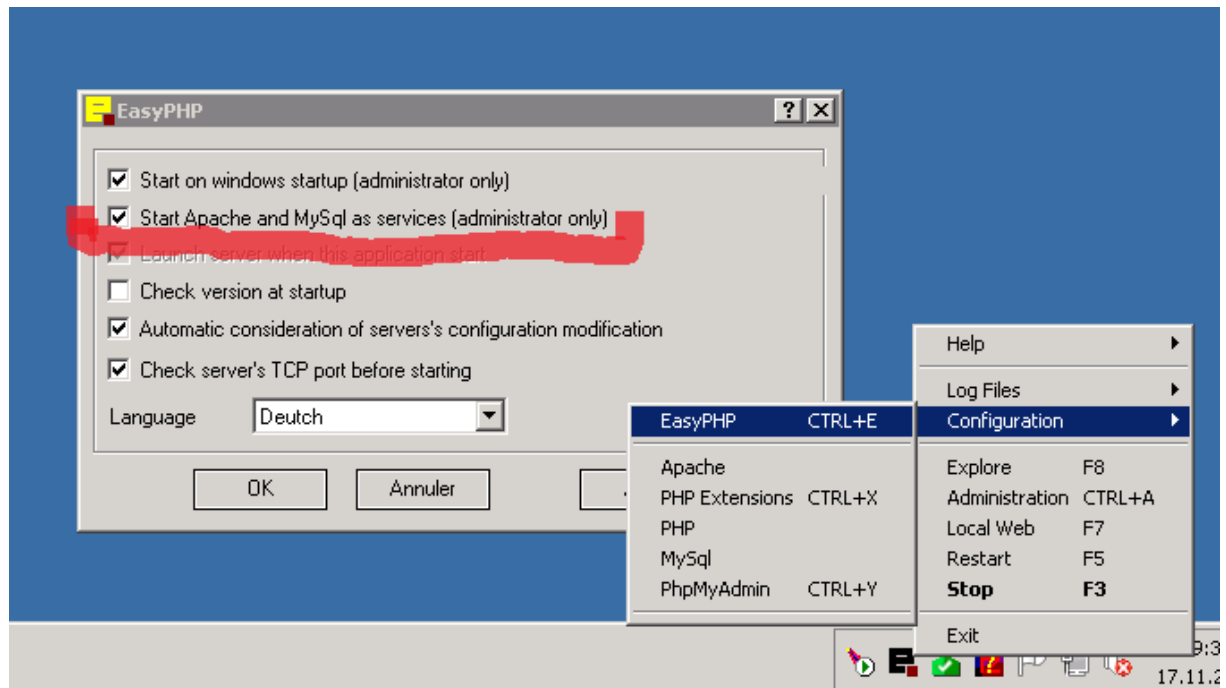
```

**Bitte der jeweiligen Konfiguration entsprechend anpassen!**

Der Import der beiden .xml Dateien geschieht im Aufgabenplaner unter Aktion -> Aufgabe importieren

9. **Optional** kann nun noch die nachträgliche Installation von Apache und mySQL aus dem EasyPHP Paket als Dienste vorgenommen werden.

Da das EasyPHP Paket mit Apache und mySQL Server nur laufen wenn EasyPHP von einem angemeldeten Benutzer gestartet wird, und nicht wie ein normaler Server immer verfügbar ist egal ob ein Benutzer angemeldet ist oder nicht.



Mit Setzen des „Start Apache and mySQL as services“ Haken ist sichergestellt das die für die Datenbank benötigten Diesnste immer zur Verfügung stehen

Das war's.

Nun sollte die oddb2 laufen, von außen erreichbar sein und regelmäßig mit den geplanten Aufgaben bestimmte Dinge erledigen.

Viel Erfolg bei der Umsetzung

Thomas / tmm