



Kostenloses eBook

LERNEN

Idap

Free unaffiliated eBook created from
Stack Overflow contributors.

#Idap

Inhaltsverzeichnis

Über.....	1
Kapitel 1: Erste Schritte mit ldap	2
Bemerkungen.....	2
Examples.....	2
PHP für die Arbeit mit LDAP einrichten.....	2
Credits	6



You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [ldap](#)

It is an unofficial and free ldap ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official ldap.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Kapitel 1: Erste Schritte mit Idap

Bemerkungen

In diesem Abschnitt erhalten Sie einen Überblick darüber, was LDAP ist und warum ein Entwickler es verwenden möchte.

Es sollte auch alle großen Themen in Idap erwähnen und auf die verwandten Themen verweisen. Da die Dokumentation für Idap neu ist, müssen Sie möglicherweise erste Versionen dieser verwandten Themen erstellen.

Examples

PHP für die Arbeit mit LDAP einrichten

Nachdem Sie Ihren LDAP-Server ordnungsgemäß konfiguriert haben, möchten wir uns jetzt verbinden. Zum Beispiel mit PHP.

- DN = Distinguished Name. Dies bedeutet, in welchem Teil der Datenbank arbeiten Sie. Kann ein Benutzer oder eine Gruppe sein (oder sogar Konfigeinstellungen).
- Eintrag: eine Entität, zum Beispiel ein Benutzer.
- Attribut: etwas innerhalb eines Eintrags, z. B. Name, Telefonnummer und E-Mail-Adresse.

Anschließen

Zuerst definieren wir Folgendes:

```
$server = "server.example.com"; //this is the LDAP server you're connecting with
$ds = ldap_connect("ldaps://$server", 636); //always connect securely via LDAPS when possible
```

Wir sind jetzt verbunden. Als nächstes wollen wir dem Server mitteilen, dass wir eine vertrauenswürdige Person sind. Die einfachste Lösung ist, den Root-DN ODER einen vorhandenen Benutzer mit den erforderlichen Berechtigungen zum Anzeigen der Datenbank zu verwenden (wahrscheinlich kann dies jeder Benutzer in der Datenbank standardmäßig tun). Wir authentifizieren uns mit der Bindefunktion.

Optionen einstellen

Zuerst erklären wir diese Optionen. Abhängig von Ihrer Serverkonfiguration können Sie dies weglassen.

```
ldap_set_option($ds, LDAP_OPT_PROTOCOL_VERSION, 3);
ldap_set_option($ds, LDAP_OPT_REFERRALS, 0);
```

Bindung

Angenommen, Sie verwenden admin und das Kennwort lautet pass123notsafe

```
$dn = "uid=admin,cn=users,dc=server,dc=example,dc=com";  
$pass = "pass123notsafe";  
$ldapbind = ldap_bind($ds, $dn, $pass); //this is the point we are authenticating
```

Das ist sehr schön. Wir sind dabei. Jetzt können wir viele verschiedene Operationen ausführen. Zum Beispiel können wir Benutzer und sogar Gruppen durchsuchen, lesen, schreiben und ändern.

Suchen

Stellen Sie sich vor, wir möchten alle Mitglieder der Gruppe "Benutzer" anzeigen.

```
$dn = "cn=users,dc=server,dc=example,dc=com"; //very important: in which part of your database  
are you looking  
$filter = "uid=*"; //don't filter anyone out (every user has a uid)  
$sr = ldap_search($ds, $dn, $filter) or die ("bummer"); //define your search scope  
  
$results = ldap_get_entries($ds, $sr); //here we are pulling the actual entries from the  
search we just defined  
var_dump($results); //will give you all results in array form.
```

Sie können foreach-Schleifen verwenden, um die Daten auf nette Weise anzuzeigen.

Okay, hab das verstanden? Fahren Sie nun mit einer gewissen Filterung fort. Wir möchten nur Benutzer in der Gruppe "Fahrradbesitzer" anzeigen, die eine E-Mail-Adresse registriert haben. Es ist wichtig zu wissen, dass sich *alle* Benutzer in cn = users befinden. Daneben können sie auch Mitglied anderer Gruppen sein.

```
//did the connecting and binding  
  
$dn = "cn=bikeowners,cn=groups,dc=server,dc=example,dc=com"; //note the extra "cn=groups" for  
looking in a group that is not "users"  
$filter = "email=*"; //email address must be set but can be anything  
$sr = ldap_search($ds, $dn, $filter) or die ("bummer"); //define your search scope
```

ldap_get_entries nun mit ldap_get_entries .

Effizienz beim Eintragen

Ein Beispiel, bei dem Effizienz angesprochen wird, das bei großen Datenbanken mit vielen Attributen pro Eintrag immer wichtiger wird. Insbesondere beim Speichern von Bildern im Attribut jpegphoto kann die Ladezeit erheblich verringert werden, wenn Sie Ihre Einträge selektiv ziehen.

Stellen Sie sich vor, wir möchten nach Benutzern suchen, die sich in der Gruppe "Fahrradbesitzer" befinden. In diesen Einträgen sind viele Informationen gespeichert. Angenommen, sie haben die folgenden Attribute: cn, uid, name, displayname, mail, Initialen, Mobiltelefon, Telefonnummer, Straße, Postanschrift, Postleitzahl und JPEG-Foto. Jetzt brauchen wir nur noch ihre UID, Namen und Initialen.

Dazu verwenden wir einen optionalen vierten Parameter von ldap_search:

```
$dn = "cn=bikeowners,cn=groups,dc=server,dc=example,dc=com";
$filter = "uid=*"; //
$justthese = array("uid", "name", "initials");
$sr = ldap_search($ds, $dn, $filter, $justthese) or die ("bummer"); //define your search scope
```

Wenn Sie `ldap_get_entries` erhalten Sie nur diese Daten.

Erweiterte Filterung

Natürlich können Sie mit LDAP eine ganze Datenbank abrufen und diese in PHP weiterverarbeiten. Wie bei MySQL ist es jedoch viel effizienter, die Verarbeitung auf der Serverseite durchzuführen. Um dies zu demonstrieren, können wir einen erweiterten Filter verwenden.

Nehmen Sie eines der obigen Beispiele als Ausgangspunkt, ändern Sie jedoch den `$`-Filter gemäß der folgenden Zeile. In unserem Beispiel möchten wir die Daten der aktiven Benutzer anzeigen. Normalerweise wird das Attribut `shadowexpire` verwendet, um diese Informationen zu speichern. Dies kann sich jedoch bei verschiedenen LDAP-Systemen unterscheiden. Wir möchten nicht nur die aktiven Benutzer anzeigen, sondern *auch* ihren Namen müssen mit einem "a" beginnen und sie müssen in Amsterdam leben.

Grundsätzlich wollen wir drei Dinge tun:

1. Am einfachsten: muss in Amsterdam leben. In diesem Beispiel wird der Wohnort im Attribut "Postanschrift" gespeichert.

```
$filter= "postaladdress=Amsterdam";
```

2. Name muss mit einem "a" beginnen. In diesem Beispiel setzt sich die UID aus dem Namen zusammen.

```
$filter= "uid=a*";
```

3. Benutzer muss aktiv sein Dieser Wert wird im Standardattribut `shadowexpire` mit dem Wert `-1` gespeichert. Je nach Ihrer Serverkonfiguration kann `shadowexpire` eine Vielzahl von Werten enthalten, sogar Datumsangaben sind möglich. Wenn ein Benutzer inaktiv ist, ist `shadowexpire 1`. Um sicherzugehen, dass wir alle Benutzer mit Ausnahme derjenigen, die wirklich inaktiv sind, erhalten, filtern wir nicht nach `shadowexpire = -1`. Stattdessen sagen wir, wir *wollen nicht*, dass sie inaktiv sind.

```
$filter= "(!(shadowexpire=1))"; //NOT is represented with "!"
```

Nun der interessanteste Teil: Kombiniere alle drei Beispiele. Wir können dies mit Klammern, AND, OR und NOT-Ausdrücken tun

```
$filter= "(&(postaladdress=Amsterdam)(uid=a*)(!(shadowexpire=1)))";
```

Schließlich könnten wir eine ODER-Anweisung mit "|" einbauen, zum Beispiel, wenn alle Benutzer

mit "a" oder "b" beginnen sollen.

```
$filter= "(|(uid=a*)(uid=n*))";
```

Sie können endlos kombinieren und beeindruckende Filter bauen. Probieren Sie es aus!

Erste Schritte mit ldap online lesen: <https://riptutorial.com/de/ldap/topic/5576/erste-schritte-mit-ldap>

Credits

S. No	Kapitel	Contributors
1	Erste Schritte mit ldap	Community , SJDS