بسمه الله الرحمن الرحیم

راهنمای LDAP

ترجمه:

حسین رستگار مقدم

مقدمه:

متن زیر حاصل ترجمه سریعی از منابع آموزشی ال دپ در اینترنت می باشد. می توانید این منابع را در انتهای متن ببینید. چند نکته درباره این متن قابل توجه است. قسمت هایی که از ترجمه حذف شده اند داخل دو خط نقطه چین قرار دارند. در صورت نیاز می توانید با مراجعه به مراجع اقدام به خواندن آنها به زبان اصلی کنید.این متن به دلیل کوتاه بودن فصل بندی نشده است . اما به دلیل اینکه در متن اصلی بعضی موارد اشاراتی به فصل های قبل و بعد در متن آمده با لینکی به آن فصل ها در متن اصلی ترجمه شده اند. به دلیل اینکه ترجمه از دقت کافی و فرصت لازم برخوردار نبوده و فرصتی برای بازبینی نداشتم ممکن است اشکلاتی در آن مشاهده کنید. راه اندازی ال دپ وقت چندانی از شما نمی گیرد و نیازی نیست تمام مطالب این متن را بخوانید تا بتوانید یک سرور ال دپ برای تمرین و تست راه اندازی کنید. همچنین شما می توانید از بسیاری از ابزار ها که روند کار را برای شما تسریع می کنند استفاده کنید اما در متن تنها از ابزار های اولیه ای که در شل لینوکس به آنها دسترسی دارید استفاده شده است. ابزار های گرافیکی بیشتر در انتهای متن همراه با مراجع استفاده شده در این آموزش آمده است، اما برای اینکه بتوانید برداشت صحیحی از نحوه کار ال دپ داشته باشید نیاز است که جزییات عملکرد آن و ساختارهای آن را بدانید. هر زمان که نیاز دیدید خواندن را متوقف کنید و شروع به برپایی نسخه اولیه ای از ال دپ کنید.

تقدیم به شهدای هسته ای

با تشکر از گروه نرم افزاری نقطه که اجازه انتشار این مطلب را دادند

h.rastegar@chmail.ir **http://www.molavy.com**

LDAP چیست:

تعریف سرویس دایرکتوری:

سرویس دایرکتوری نرم افزاری است که مسئول چگونگی ذخیره کردن، بازیابی، دسترسی و سازماندهی اطلاعات در یک دایرکتوری است. در مهندسی نرم افزار دایرکتوری یک مپ بین نام ها و مقادیر است.

یعنی با دادن یک نام به شما مقداری را بر می گرداند یا برای شما جستجو می کند.

درست مثل یک فرهنگ لغت که یک نام ممکن است حاوی معنی ها و تعاریف متعددی باشد. یک نام در یک دایرکتوری ممکن است حاوی قطعه های متعددی از داده ها باشد. باز در فرهنگ لغت یک کلمه ممکن است در مکان های مختلف معانی مختلفی داشته باشد، به همین ترتیب یک نام در دایرکتوری ممکن است حاوی انواع مختلفی از داده ها را برگرداند.

به صورت تئوری، ال دپ یک پروتکل است که این کار را انجام می دهد:

تعریف روشی که توسط آن دایرکتوری داده قابل دسترس است.

همچنین لزوما تعریف میکند و توضیح میدهد که داده ها چگونه در دایرکتوری سرویس نمایش داده شوند.(مدل داده) در نهایت ال دپ مشخص میکند که چگونه داده ها به دایرکتوری سرویس وارد و چگونه از آن خارج شوند (با استفاده از LDIF).

ال دپ کاری به این ندارد که داده ها چطور ذخیره میشوند و چگونه تغییر میکنند

ال دپ چهار مدل تعریف می کند:

Information model:

دیتا مدل تعریف میکند چگونه داده ها و اطلاعات در سیستمهای مبتنی بر ال دپ نمایش داده شوند.

Naming model:

این چیزی است که شما در همه جای ال دپ با آن سر و کار دارید، بنابراین در همین ابتدا آن را توضیح خواهیم داد.

Functional model:

وقتی شما میخوانید، می نویسید، جستجو میکنید از این فانکشنال مدل استفاده می کنید.

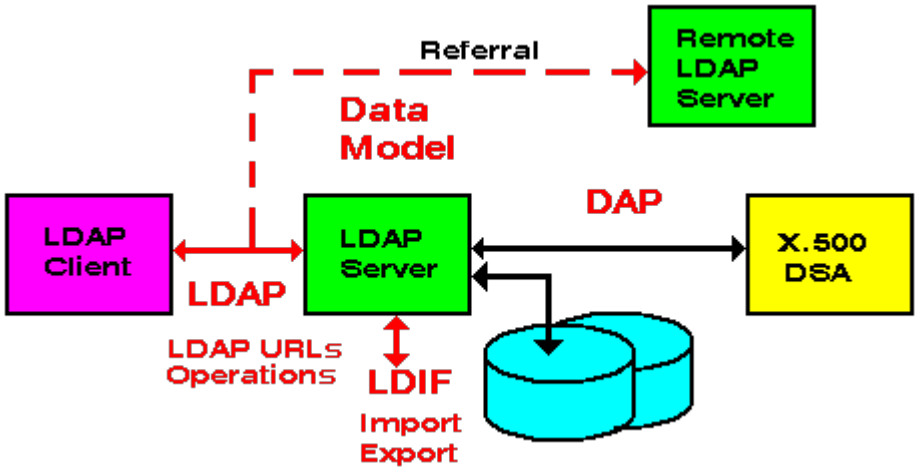
Security Model:

میتوانید با استفاده از آن به صورت جز به جز تعریف کنید که چه کسی میتواند چه کاری بر روی چه فایلی انجام دهد.

این موضوع کمی پیچیده ولی بسیار قدرتمند است. برای شروع امنیت را فراموش کنید.

شما تقریبا همیشه میتوانید برگردید و موضوعات امنیتی را اصلاح کنید. هرجا مشکلی از این لحاظ بود در متن قید می شود.

تصویر زیر محدوده فعالیت و نمونه کار ال دپ را نشان می دهد.



هر بخش در جای خود توضیح داده خواهد شد اما در ابتدا به چند نکته اساسی باید توجه کرد.

۱) ال دپ کاری به این ندارد که داده ها چگونه ذخیره می شوند .فقط چگونگی دسترسی به آنها را مشخص می کند. فقط اکثر برنامه های ارائه سرویس ال دپ یک پایگاه داده استاندارد برای کارهای خودشان دارند به جز openLDAP که اجازه انتخاب پایگاه داده مختلفی را برای این موضوع می دهد.

۲) وقتی شما به ال دپ صحبت میکنید کاری به این ندارید که داده ها از کجا می آیند .تمام نکته همین است که شما این مرحله را پنهان می کنید .ممکن است داده ها از چند سرور ال دپ محلی بیایند یا از یک سرویس دهنده X.۵۰۰ باشند.

۳)این دو موضوع را کامل در ذهنتان از هم جدا کنید.

دسترسی به یک سرویس ال دپ

عملیات های یک سرویس ال دپ

در هنگام طراحی یک سیستم ال دپ، به این فکر کنید که چه کاری از آن میخواهید که انجام دهد. (شماتیک و سازماندهی داده ها) و نمونه پیاده سازی را فراموش کنید. در مرحله دوم باید به این فکر کنید که داده ها کجا باشند و چطور و کجا میخواهید آنها را ذخیره کنید. (در زمان تنظیم کردن ال دپ)

۴)بعضی مؤسسات تجاری برنامه های با قابلیت نمایش ال دپ از پایگاه داده ی اس کیو الی نوشته اند.

قسمتهای حذف شده

.....

۳.۲ LDAP vs. Database

Read Optimized

.....

نگاهی به سازماندهی داده ها

در استفاده های ساده از ال دپ ممکن است مدل داده هایی که خوانده میشوند با آدرس دهی فیزیکی آنها یکی باشد. حال پیشفرض را بر این میگذاریم که آبجکت مدل داده ما از حالت فیزیکی ذخیره سازی داده ها هیچ اطلاعی ندارد. در حقیقت تمام سادگی ال دپ در این نکته پنهان شده است.

این ویژگی در مقابل ساختار ای کیو ال کوئری بیان میشود که اطلاعات کاملی از ساختار داده جداول و ارتباطات آن وجود دارد.

.....

Data Synchronisation

۱.۳.۲ LDAP Usage Summary

.....

مدل داده (آبجکت) ال دپ

سیستمهای مبتنی بر ال دپ از مدلی از داده ها استفاده میکنند که فرض میکند و نشان میدهد داده ها در ساختار درختی از آبجکت ها هستند.

این موضوع به معنی آن نیست که ال دپ مانند پایگاه داده های اس کیو ال است .همانطور که در بالا توضیح داده شد ال دپ به این کاری ندارد که داده ها واقعا چگونه ذخیره شده اند .تعریفی از این موضوع ندارد

اما عملیات های پایه مانند تغییر، حذف و خواندن داده ها را به صورت آبجکت در نظر میگیرند. (اغلب)

ساختار درختی آبجکت ها

داده ها در سیستمهای مبتنی بر ال دپ به صورت یک ساختار درختی از آبجکت ها تعریف می شوند.

هرکدام به یک مدخل(Entry) خوانده می شوند .در مجموع کل ساختار درختی DIT نامیده میشود.(data information tree)

بالاترین نقطه یک DIT را Root می نامیم (baseو suffix نیز نامیده می شوند)

هر مدخل در درخت یک مدخل والد دارد (parent)و صفر،یک یا بیش از یک مدخل فرزنددارد (آبجکت)

هر مدخل فرزند برادر هایی دارد که والد آنها یکسان است.

هر مدخل از چند ObjectClass ساخته شده است.

Objectclassهاوی صفر یا بیشتر attribute هستند.

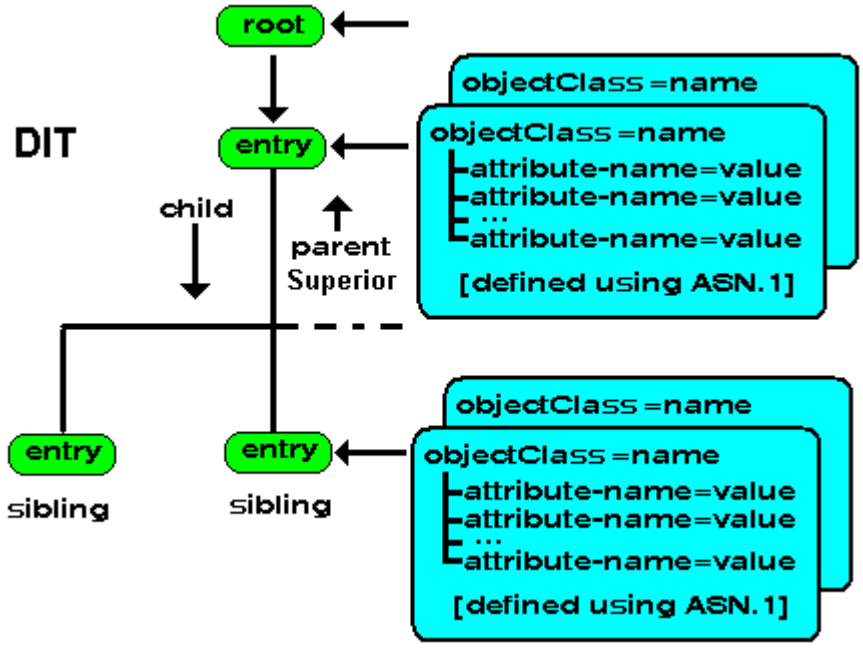
و بالاخره attribute ها دارای نام(بعضی از نام ها اختصاری و بعضی Alias هستند)و اغلب حاوی دیتا هستند.

ویژگیهای ObjectClass و attribute ها در ASN.1 شرح داده شده است.

خوب تا اینجا شما همه چیز را درباره ال دپ می دانید.

سایر موضوعات جزئیات کار است اما مباحث کلیدی و اصلی گفته شده است.

دیاگرام زیر تمام روابط گفته شده را نشان می دهد.



یک بار دیگر به طور خلاصه:

۱) هر مدخل از یک یا چند Objectclass ساخته شده است

۲) هر Objectclass یک نام دارد

۳) هر attribute یک نام دارد. اغلب حاوی داده است و عضوی از یک Objectclass می باشد

Attributes(اتریبیوت ها)

هر اتریبیوت یک نام دارد و اغلب حاوی داده است. هر اتریبیتوت عضو یک یا چند آبجکت کلاس است .اتریبیتوت ها شامل چند ویژگی به خصوص است.

۱) هر اتریبیوت عضو یک یا چند Objectclass می باشد.

۲) هر اتریبیوت نوع دادهای (data type) که میتواند داشته باشد را تعریف می کند.

۳) یک اتریبیوت ممکن است عضو اجباری یا اختیاری یک آبجکت کلاس باشد .یک اتریبتوت واحد ممکن است در یک آبجکت کلاس اختیاری و در دیگر آبجکت کلاس اجباری باشد.

این آبجکت کلاس است که در مورد این خصیصه صحبت میکند

۴) در هر سطحی از درخت یک محتوا برای یک اتریبویت میتواند به عنوان ویژگی یکه آن واحد استفاده شود .هر خصیصه ای از اتریبیتوت را میتوان با این ویژگی شناخت . حتی داده یک اتریبیتوت میتواند دو اتریبیتوت دیگر باشد.

ObjectClasses

آبجکت کلاسها دسته ای از اتریبوت ها هستند.

به صورت پیشفرض تعدادی آبجکت کلاس تعریف شده است .ولی برای کار مناسب نمی باشند و شما باید آبجکت کلاسهای خود را تعریف کنید

آبجکت کلاس ها مشخص میکنند که اتریبوت های آن الزامی باشد یا اختیاری باشد.
آبجکت کلاسها ممکن است در ساختار درختی تمام خصیصه های خود را از والد به ارث برند

تعریف درخت و افزودن اطلاعات

تعریف ساختار درختی و آغاز و داده های اولیه با افزودن مدخل شروع می گردد (با آبجکت کلاس ها و اتریبیوت های مربوطه)

در آغاز از رووت شروع می کنیم به پایین ساختار درختی میرسیم بنابراین یک والد باید همیشه قبل از اینکه فرزند بتواند به وجود بیاید افزوده شود

افزودن فایلها به چند روش میتواند انجام شود .یکی از آنها با استفاده از LDIF انجام میشود که در فصل های بعدی به تفضیل بحث خواهد شد.

LDIF یک فایل متنی است که ساختار سلسله مراتبی اطلاعات را تعریف میکند و داده هایی که باید به هر اتریبیوت اضافه
گردد. متن زیر محتوای یک فایل LDIF است که رووت را با تعریف (dc=example,dc=com)تعریف کرده و یک

مدخل فرزند در زیر people ایجاد می کند.

لازم نیست تمام تعاریفی که این فایل انجام میدهد را در این مرحله بدانید.
در فصل Chapter ۵ (samples) توضیحات مفصلی از این موارد آورده شده است.

برای این مرحله تنها کافیست بدانید LDIF میتواند برای برپایی DIT استفاده شود و محتوای آن چیزی شبیه اطلاعات زیر است.

۱:version

version not strictly necessary but good practice to include for future

DEFINE DIT ROOT/BASE/SUFFIX

۷۷ format ۳۲### uses RFC

dcObject is an AUXILIARY objectclass and MUST

have a STRUCTURAL objectclass (organization in this case)
this is an ENTRY sequence and is preceded by a BLANK line

dn: dc=example,dc=com

dc: example

description: The best company in the whole world objectClass: dcObject

objectClass: organization o: Example, Inc.

FIRST Level hierarchy – people

this is an ENTRY sequence and is preceded by a BLANK line

dn:ou=people, dc=example,dc=com

ou: people

description: All people in organisation objectClass: organizationalUnit

SECOND Level hierarchy – people entries

this is an ENTRY sequence and is preceded by a BLANK line

dn: cn=Joe Schmo,ou=people,dc=example,dc=com objectclass: inetOrgPerson

cn: Joe Schmo

sn: Schmo

uid: jschmo

mail: joe@example.com

mail: j.schmo@example.com
ou: sales

نکته مهم:

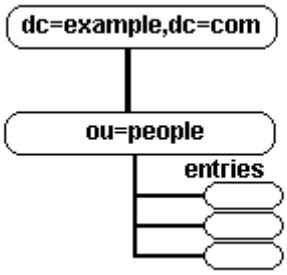
خطی آغازینی که در LDIF با :dn آغاز میشود به سرور میگوید که مدخل را در کجای ساختار درختی قرار دهد.تا زمانی که

dnیکه باشد مهم نیست که attribute حاوی چه داده ای باشد.

مثال بالا مدخل نهایی از cn=Joe Schmo استفاده کرده است، میتوانست از هر مقداری مثل uid=jschmo استفاده کند.

جستجوی ال دپ میتواند به هر نوع ترکیبی از اتریبوتها انجام شود و میتواند صرف نظر از:dn که آنرا ایجاد کرده است پیدا کند.

اگر مدخل برای استفاده در احزار هویت به کار میرود خیلی مهم است که در اینجا :dn و مکان bind کردن آن خیلی مهم می شود. بعدا LDIF را توضیح خواهیم داد اما LDIF بالا ساختار زیر را به وجود می آورد.



به محض ایجاد و راه اندازی یک DIF ، داده ها بعدی را میتوان با DIF دیگر، یک LDAP Browser از وب یا هر برنامه ای دیگری به آن اضافه کرد.

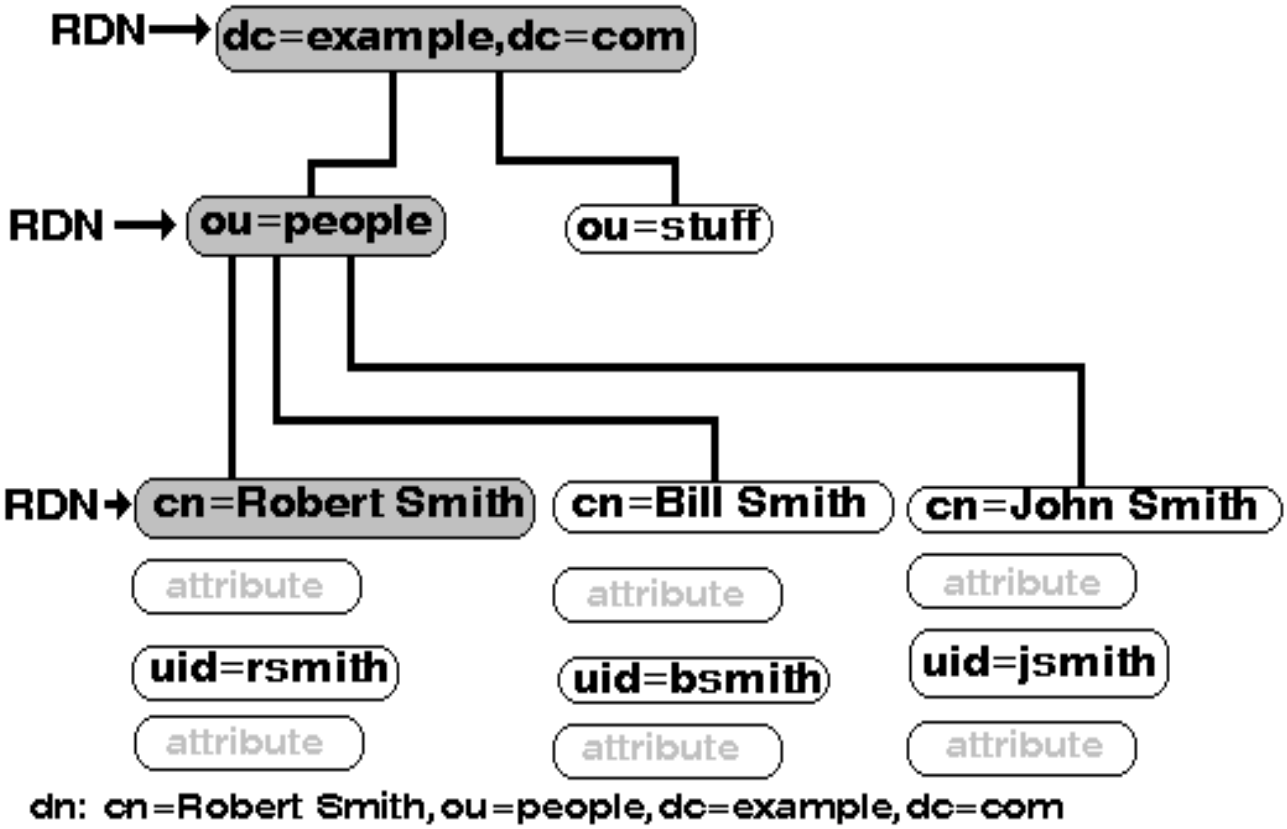
داده ها میتوانند با LDIF اکسپورت یا بکآپ گرفته شوند

حرکت و پیمایش در DIF

بعد از ورود اطلاعات در دایرکتوری ما میخواهیم از آنها استفاده کنیم

برای این کار ما نیاز داریم که فرمان ارسال کنیم (خواندن ،جستجو، تغییر و...) برای این کار ما باید به ال دپ بگوییم که داده ها کجا هستند (برای نوشتن) یا تقریبا کجا هستند (برای جستجو) در حقیقت ما نیاز داریم که در درخت پیمایش کنیم. در قسمت قبل گفتیم که هر اتریبیتوت در هر سطح باید شامل حداقل یک داده یکه باشد تا آن را از سایر اتریبیتوت ها مجزا کند. با ایجاد مسیر هایی که از این نوع اتریبیوت ها ما میتوانیم به مدخل دلخواه یا نقطه جستجوی آغازین خود دست یابیم.

این مسیر ها به اصطلح Distinguished Names یا اختصارا dn شناخته می شوند. هر داده یکه ای که در زیر این مسیر قرار گیرد اصطلاحا Relative Distinguished Name یا به اختصار RDN شناخته می شود. دیاگرام زیر DN و RDN ها را شرح می دهد.



توضیحات بیشتر همراه با مثال های کاربردی

.....

۲.۵ LDAP Referrals and Replication

۲.۵.۱ LDAP Referrals

۲.۵.۲ LDAP Replication

.....

ال دپ شم،آبجکت کلاس و اتریبیوت

به دلیل اینکه آبجکت ها و کلاسها و اتریبیوت ها به هم تنیده هستند ما از واژه ستاف (stuff) در مجموع استفاده می کنیم. شما میتوانید هر نام دیگری که خواستید را استفاده کنید.

وقتی شما یک اینتری(مدخل) در یک دیت ایجاد میکنید، داده های آن در قالب اتریبیوت ها ذخیره می شوند. که این اتریبیوت ها در آبجکت کلاس ها جمع میشوند و در مجموع در شم ها ذخیره می شوند.

قدرت و پیچیدگی ال دپ در این است که میتواند حجم زیادی از اتریبیوت ها و آبجکت کلاسهای پراکنده را لود کند.

بررسی یک ستاف ال دپ

همه چیز در ال دپ درختی است، این شامل حال ابجکت کلاسها و اتریبیوت ها نیز می شود.

شم ها مهم هستند.آنها تنها آبجکت کلاسها و اتریبیوت های مرتبط را به صورت بسته در می آورد.

۱) شم ها یک بسته یک بسته ساده هستند:

تمام آبجکت کلاسها و اتریبیوت ها در داخل شم ها تعریف می شوند (چند آبجکت کلاس و اتریبیوت ویژه عملیات ها در داخل خود سرور الدپ تعریف شده اند و نیاز به تعریف ندارند اما در اینجا از آنها صرف نظر می کنیم)

تمام شم هایی که دربر دارنده آبجکت کلاسها و اتریبیوت ها هستند و در ال دپ میخواهیم از آنها استفاده کنیم باید توسط ال دپ سرور شناخته شده باشند. این موضوع با include کردن در فایل (slapd.conf فایل تنظیمات سرور ال دپ) انجام می شود.

یک اتریبیوت تعریف شده در هر یک از شم ها میتواند در هر آبجکت کلاس در داخل شم های دیگر استفاده شود.

۲)یک آبجکت کلاس گروهی از اتریبیوت ها است:

آبجکت کلاس در داخل شم ها تعریف می شوند.

آبجکت کلاسها ممکن است به صورت درختی تعریف شوند .در این حالت آنها تمام ویژگیهای والد خود یا جداول (ریشه اول) را به ارث می برند.

آبجکت کلاس ها ممکن است به ۳ صورت باشند:

ساختاری : (Structural) در این حالت آنها برای ایجاد مدخل ها به کار (رفته اند)آبجت های داده.(
معین : (AUXILIARY) در این حالت میتوانند به هر مدخل مناسب اضافه شوند.

انتزاعی از چیزی که وجود ندارد باشند : (ABSTRACT) شایع ترین آبجکت کلاس انتزاعی TOP است که برای تعریف بالاترین سطح یک آبجکت کلاس استفاده میشود و هر ساختار درختی را پایان می دهد.

اگر یک آبجکت کلاس از یک درخت باشد (معین یا ساختاری) حتما باید همنوع آبجکت کلاس جداول خود باشد
استثناء این حالت این است که آبجکت کلاس جداول از نوع انتزاعی باشد.

آبجکت کلاس وقتی معنی میدهد که حاوی اتریبیوت باشد (در اصلاح آن را نگهدارنده اتریبیوت می گویند).
آبجکت کلاس تعریف میکند که یک اتریبیوت برای آن الزامی باشد یا اختیاری باشد.

۳)اتریبیوت ها اغلب حاوی داده هستند

هر اتریبیوت در یک یا بیشتر از یک آبجکت کلاس است

برای استفاده از هر اتریبیوت در یک مدخل، آبجکت کلاس آن باید در مدخل تعریف شده باشد و آبجکت کلاس آن باید در شم تعریف شده باشد .همچنین شم باید در ال دپ سرور تعریف شده باشد

ویژگیهای یک اتریبیوت با استفاده از نکات ASN ۱.تعریف می شوند.

یک خصیصه میتواند یک بار در هر آبجکت کلاس خود ظاهر شود (SINGLE-VALUE)و یا بیش از چند بار در آبجکت کلاس خود ظاهر شود (MULTI-VALUE).MULTI-VALUE)به صورت پیشفرض است.

یک تعریف اتریبیوت ممکن است در ساختار درختی باشد. در این حالت تمام خصوصیات والدهای خود را به ارث می برد.

برای مثال :commonName (cn), givenName (gn), surname (sn) همه فرزند های اتریبیوت Name هستند.

تعریف اتریبیوت شامل نوع و نمو (Syntax) خود نیز می شود. برای مثال نوع های عددی و رشته ای و نحوه عملکرد آنها . برای مثال یک مقایسه حساس به بزرگی و کوچکی هست یا خیر

۴) مدخل ها دسته ای از آبجکت کلاسها درون یک ساختار درختی هستند.
مدخل ها باید یک و تنها یک آبجکت کلاس از نوع ساختاری داشته باشند

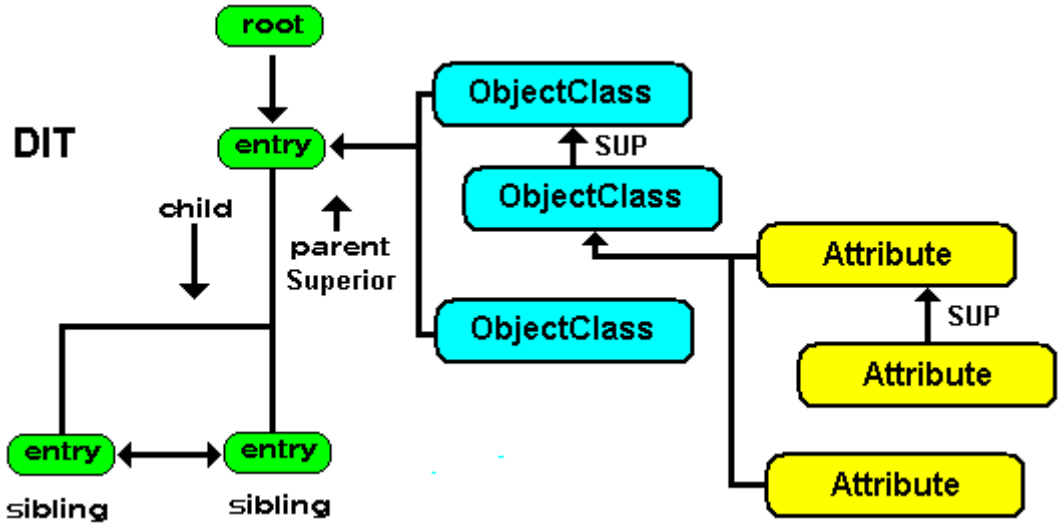
یک آبجکت کلاس ساختاری ممکن است یک والد داشته باشد (عضو ساختار درختی) که آن هم از نوع ساختاری باشد که در مجموع در ساختار هر دو به عنوان یک آبجکت کلاس ساختاری شناخته می شوند

یک مدخل میتواند هر تعداد آبجکت کلاس از نوع معین داشته باشد
یک مدخل ممکن است تنها یک آبجکت کلاس از نوع انتزاعی داشته باشد

مدخل ها میتوانند حاوی مدخل فرزند باشند و در زیر آنها در آدرس ساختار درختی ظاهر می گردند.
مدخل ها میتوانند حاوی مدخل والد باشند که در بالای آنها در آدرس ساختار درختی ظاهر می گردند.

مدخل ها میتوانند برادرانی داشته باشند که در سطح یکسان آنها در ساختار درختی ظاهر میگردند .مدخل های برادر یک والد یکسان دارند.

تصویر زیر تعدادی از روابط گفته شده را شرح می دهد.



شم ها در ال دپ

شم ها در ال دپ چیزی نیستند جز بسته ای از آبجکت کلاسها و اتریبیوت های مرتبط

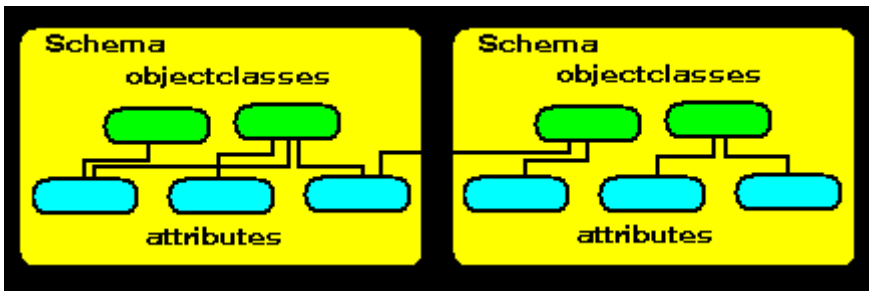
ممکن است در ابتدا تنها یک شم برای نگهداری تمام اطلاعات استفاده شود (چیزی شبیه شم های دیتابیس های رابطهای)، اما در ادامه به این شکل نیست. شما آبجکت کلاسها و اتریبیوت های مفید زیادی را در اطراف پراکنده میکنید. قدرت ال دپ در همین استفاده از اطلاعات به ظاهر آشفته است.

این یک قانون است:

برای هر اتریبیوت یا آبجکت کلس، شامل تمام آبجکت کلاسها و اتریبیوت های والد آن باید در شم تعریف شوند .و آن شم باید به ال دپ سرور شناسانده شود.

دوباره در اپن ال دپ با استفاده از کلید واژه include در فایل slap.conf میتوانید این کار را انجام دهید.

دیاگرام زیر قسمتی از توضیحات بالا را نمایش می دهد.



آبجکت کلاسها در ال دپ

آبجکت کلاسها دسته ای از اتریبیوت ها هستند که خصوصیات زیر را دارند
۱)آبجکت کلاسها در شم ها تعریف میشوند

۲)یک آبجکت کلاس ممکن است خود جز یک ساختار درختی آبجکت کلاس باشد.
در این حالت تمام ویژگیهای والد خود را به ارث می برند.

برای مثال آبجکت کلاس inetOrgPerson فرزند organizationalPerson است که خود فرزند person است، که آن نیز فرزند top است (یک آبجکت کلاس انتزاعی پایان هر ساختار درختی آبجکت کلاسها است).

۳)یک آبجکت کلاس یک نام یکه سراسری معرف دارد unique name or identifier--

۴)یک آبجکت کلاس همانطور که دارای اتریبیوت است خود نیز یک میتواند مانند یک اتریبیوت در جستجو مورد استفاده قرار گیرد.

۵) یک آبجکت کلاس تعریف میکند که اتریبیتوتش اجباری باشد یا اختیاری

۶) اگر عضو یک ساختار درختی باشد تمام ویژگیهای والد را به ارث می برد.

۷) یک یا بیش از یک آبجکت کلاس باید در یک مدخل تعریف شوند.

۸) یک و تنها یک آبجکت کلاس از نوع انتزاعی باید در مدخل ال دپ تعریف شوند

۹)(هر آبجکت کلاس ساپورت شده توسط توسط سرور ال دپ در یک کالکشن (collection)میتواند توسط ساب شم ها(subschema) کشف گردد.

تعریف یک آبجکت کلس

تعریف رسمی آبجکت کلاسها در RFC ۴.۴ ۲ ۵ ۲۲ section آمده است و به دین شکل است:

ObjectClassDescription = "(" whsp

numericoid whsp; ObjectClass identifier

["NAME" qdescrs]

["DESC" qdstring]

["OBSOLETE" whsp]

["SUP" oids]; Superior ObjectClasses

[("ABSTRACT"/"STRUCTURAL"/"AUXILIARY")whsp]

; default structural

["MUST" oids]; AttributeTypes

["MAY" oids]; AttributeTypes

whsp ")"

whsp به معنی فاصله است و باید آنجا باشد.

به جای اینکه هر پارامتر را به صورت جداگانه شرح دهیم بیایید نگاهی به چند مثال بیندازیم
آبجکت کلاس ساده استاندارد زیر از تعریف country در core.schema خود openLDAP آمده است

NAME 'country' SUP top STRUCTURAL ۲.۶.۵.۲ objectclass (

MUST c

MAY (searchGuide $ description))

حال بیایید آن را جز به جز بررسی کنیم

objectclass اصطلاحی است که به ما میگوید این یک تعریف آبجکت کلاس است.

NAME 'country'۲.۶.۵.۲

تعریف نام سراسری یکه آبجکت کلاس است که از دو بخش تشکیل شده است.

'NAME 'country به شما اجازه میدهد که با نامی قابل فهم برای انسان بتوانید از این آبجکت کلاس استفاده کنید قسمت یکه سراسری آن ۲.۶.۵.۲ است که به آن (ObjectIdentifier) OID میگویند

فرقی نمیکند که چه عددی برای چه آبجکت کلسی استفاده شود، تنها مهم این است که یکه باشد.

میتوانید یک OID واحد را برای شرکت خود با استفاده از IANA بگیرید. این خیلی بعد است که از OID که قبل استفاده شده است استفاده کنید.

SUP 'top'

تعریف میکند یک این آبجکت کلاس یک والد دارد (SUPerior) و جزئی از یک ساختار درختی است.

در این مورد والد آن top است که پایان هر ساختار درختی است. هر آبجکت کلاس میتواند یک یا چند والد آبجکت کلاس داشته باشد.

STRUCTURAL تعیین میکند که این آبجکت کلاس میتواند یک مدخل در دیت باشد.

تنها یک نوع آبجکت کلاس ساختاری میتواند در مدخل باشد اما یک آبجکت کلاس ساختاری میتواند در یک درخت باشد) و خود فرزند یک آبجکت کلاس ساختاری باشد)(.اطلاعات بیشتر در این باره. (یک آبجکت کلاس میتواند از نوع انتزاعی باشد

مثل(top)

در نهایت یک آبجکت کلاس میتواند از نوع معین باشد که به معنی آن است که حاوی اتریبیوت است و ممکن است توسط هر آبجکت کلاس ساختاری برای ایجاد اینتری استفاده شود.

اما به تنهای نمیتوانید یک مدخل را به وجود آورد

'DESC 'description

این مورد در مثال بالا نیست. به دلیل اینکه آنرا حذف کرده ایم.

همانطور که از نام آن مشخص است توضیحی در مورد محتوا و اطلاعات درون آبجکت کلاس می دهد. این مورد تنها درک بهتر برای کسانی که شم را میخوانند قرار داده شده است. آبجکت کلاس با توضیح به شکل زیر است

NAME 'country' SUP top STRUCTURAL ۲.۶.۵.۲ objectclass (

character iso assigned country code'۲ DESC '

MUST c

MAY (searchGuide $ description))

میتوان با توضیح بسیاری از ابهامات را رفع کرد.

MUST c

MUST نشان میدهد که اتریبیوت ها در این لیست اجباری هستند (در اینجا اتریبیوت c)

اگر تنها این اتریبوت مقدار دهی شود یک مورد از این آبجکت کلاس ایجاد میشود و اگر در اینجا یک آبجکت کلاس ساختاری تعریف میشد مدخل ایجاد نمی شد.

در اینجا یک مورد تعریف شده است

برای تعریف چند گانه از پرانتز استفاده میکنیم و میان هر اتریبیوت یک علامت دلار میگذاریم

به این شکل

attr۱ $ attr۲ $ attrn

اگر اتربیوت اجباری نداشته باشیم این قسمت حذف می شود.

MAY (searchGuide $ description)

MAY نشان میدهد که اتریبیوت ها در این لیست اختیاری هستند و نیازی به حضور آنها برای ایجاد یک آبجکت کلاس نیست.

یک اتریبیوت نیاز به پرانتر ندارد. اگر اتریبیوت اختیاری نداشته باشیم این قسمت حذف می شود.

چند مثال دیگر

چگونه top تعریف می شود.

.۰ NAME 'top' ABSTRACT ۶.۵.۲ objectclass (

MUST objectClass)

همچنین این مورد نشان میدهد چگونه یک آبجکت کلاس ساختاری تعریف می شود. به دلیل اینکه تاپ همیشه بالاترین قسمت یک ساختار است نمیتواند SUP داشته باشد.

نیز طبق استاندارد های گروه X.۵۰۰ تعریف شده است.OID

بسیاری اسناد تاکید دارند که top در LDIF تعریف شود .اما اجباری برای این کار نیست.

تعریف dcObject

NAME 'dcObject'۴۴۳.۶۶۴۱.۱.۴.۱.۶.۳.۱ objectclass (

۷: domain component object'۴۲۲ DESC 'RFC SUP top AUXILIARY MUST dc)

این مورد نشان میدهد چطور یک آبجکت کلاس معین تعریف کنیم .نمیتوان تنها با تعریف نوع معین یک مدخل

ایجاد کرد.

همچین در این مثال از OID خصوصی تجاری استفاده شده است(ObjectIdentifier).
کد زیر نشان میدهد چطور میتوان بر مبانی dcObject یک DN تعریف کرد:

dn: dc=example,dc=com

dc: example.com

objectclass: dcObject

objectclass: organization o: Example, Inc.

این objectclass: organization است که مدخل را ایجاد می کند DcObject .روی آن سوار می شود.

مثال زیر نحوه تعریف pilotOrganization را نشان میدهد و همچنین نشان میدهد چطور یک آبجکت کلاس میتواند بیش از یک والد داشته باشد.در این مثال هر دو organization a و organizationalUnit والد های pilotOrganization هستند.

در این مثال فرزند خصوصیات خود را از تمام والدین به ارث می برد)اطلاعات بیشتر در مورد خواص توارث در ال دپ(

objectClasses:(۰ NAME 'pilotOrganization'۲.۴ ۰۰.۱ ۰۰.۳ ۰۰ ۲۹۱.۲۴۳۲.۹ ۰.

SUP (organization $ organizationalUnit) STRUCTURAL

MAY buildingName)

ما اصطلحات OBSOLETE را حذف کرده ایم .اگر شما این اصطلاحات را دیدید یعنی نباید از آبجکت کلاس استفاده کنید.

اتریبیوت ها در ال دپ

اتریبیوت ها اغلب حاوی اطلاعات هستند و ویژگیهای زیر را دارند:

موردی که قبل گفته شد در مورد خصوصیات اتریبوت ها در ال دپ()۸

تعریف یک اتریبیوت

شیوه رسمی تعریف یک اتریبیوت بر اساس)۲.۴RFC) ۲۲ ۵ ۲ section مشخص میگردد و شبیه به زیر است.

AttributeTypeDescription="(" whsp

numericoid whsp; AttributeType identifier

["NAME" qdescrs]; name used in AttributeType

["DESC" qdstring]; description

["OBSOLETE" whsp]

["SUP" woid]; derived from this other

; AttributeType

["EQUALITY" woid; Matching Rule name

["ORDERING" woid; Matching Rule name

["SUBSTR" woid]; Matching Rule name

["SYNTAX" whsp noidlen whsp] ; Syntax OID

["SINGLE-VALUE"whsp]; default multi-valued

["COLLECTIVE" whsp]; default not collective

["NO-USER-MODIFICATION"whsp]; default user modifiable

[X-ORDERED whsp type]; non-standard - default not X-ORDERED

["USAGE" whsp AttributeUsage]; default userApplications
whsp ")"

کاراکتر whsp به معنای فاصله است و باید وجود داشته باشد .به جای توضیح جز به جز هر بخش بیایید باز با چند مثال شروع کنیم:

مثال زیر یک تعریف استاندارد commonName (cn) است که از فایل core.schema خود openLDAP برداشته شده است.

attributetype (۲.۵.۴.۳ SUP name) 'cn' 'commonName') NAME

توضیح جز به جز مثال بال:

attributetype مشخص میکند یک این یک تعریف اتریبیوت است.

NAME ('cn' 'commonName')۲.۵.۴.۳

یک نام سراسری برای این اتریبیوت تعریف میکند و از دو قسمت تشکلی شده است

NAME ('cn' 'commonName')

برای تعریف یک نام قابل فهم برای انسان به کار می رود.در این مثال شما یک نام کامل دارید و یک نام کوتاه .(cn)در اینجا هیچ محدودیتی برای نام های اختصاری و کوتاه وجود ندارد،البته تا زمانی که تکراری نباشند .برای تعریف چند نام با فاصله آنها را از هم جدا کنید و همه را درون پرانتز قرار دهید .به خاطر اینکه cn اول آمده است آنرا اولیه (primary)می نامند.
این وقتی مدخل در ایندکس ها میآید خیلی مهم است.

'name' SUPنشان میدهد که این اتریبیوت یک والد دارد و عضو یک ساختار درختی است .در اینجا والد آن name

است و این اتریبیوت تمام ویژگی هایش را از آن به ارث میبرد به علوه اینکه یک سری ویژگی مخصوص به خود دارد. SUBمیتواند یک OID یا یک نام باشد.اگر به جای name عدد آن را می آوردیم هیچ فرقی نمیکرد و فقط کمتر قابل فهم می شود.

تعریف زیر name را تعریف می کند)والد(cn

NAME 'name'۱۴.۴.۵.۲ attributetype (
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX ۱.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ ۵ ۲۳ ۷ ۸۶)

توضیح بخشهای جدید

EQUALITY caseIgnoreMatch

نشان میدهد که چگونه این و فرزند هایش در هنگام فیلتر های جستجو عمل می کنند. در این حال اصطلح روبه روی آن باعث عدم حساس بودن به کوچک و بزرگ بودن حروف می شود.

CaseIgnoreMatch یک قانون همانند یابی است و در ساب شم ها تعریف میشود

SUBSTR caseIgnoreSubstringsMatch مانند بالا اما اینبار در جستجو های ساب استرینگ استفاده می شود)جکس ها مثل ((cn=jim) که اینجا غیر حساس بودن به حروف کوچک و بزرگ را تعریف کرده و همانند بالا یک قانون همانند یابی است و در ساب شم ها تعریف می شود. SYNTAX ۱.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ ۸۶۷۲۳۵

این یک OID است مشخص میکند چه نوع دیتای را این نوع اتریبیوت قبول میکند و چه ارزیابی هایی باید روی آن قبل از ذخیره انجام شود.

لیست کامل در اینجا ۲.۳.۴ section ۲ ۵ ۲۲.RFC

در اینجا تعریف کرده است که دایرکتوری رشته ای که در section ۲ ۵ ۲۲ RFC ۱.۶ ۰ تعریف شده است و ۸ UTF- طبق استاندارد ISO ۱ ۰ ۶۴۶ مقدار} ۸۶۷۲۳ {نشان دهنده حداکثر مقدار قابل پذیرش آن است و اختیاری می باشد.
.....

Other Characteristics

SINGLE-VALUE

USAGE 'AttributeUsage'

ORDERING 'matchingrule'

Additional Definition Elements:

'X ORDEREDtype'

.....

قوانین همانند یابی

قوانین همانند یابی به قسمت عملیاتی ال دپ سرور مربوط می شود.
قوانین همانند یابی به شیوههای مقایسهای که در ال دپ سرور وجود دارد مربوط می شود.
۱(قوانین همانند یابی در داخل خود سرور ال دپ تعریف شدهاند و نیاز به تعریف مجزا ندارند.
matchingrulesدر داخل subschema تعریف شده اند. ۲(قوانین همانند یابی در مجموعهای به نام ,EQUALITY
SUBSTRو ORDERING در داخل اتریبیوت تعریف می شوند.اگر جستجو ۳(قوانین همانند یابی توسط نتواند از ویلکارد ها استفاده کند یعنی SUBSTR تعریف نشده است

.....

Defining matchingRule

.....

قوانین همانند یابی داخلی openLDAP

استفاده از دستوری مانند زیر میتوانید قوانین همانند یابی اپن ال دپ را که در ساب شم ها است ببینید

ldapsearch -H ldap://ldap.example.com -x -s base -b"cn=subschema" (objectclass=*)" matchingrules

ldap.example.comرا با هاست ال دپ خود جایگزین کنید و اگر از سرور لوکال استفاده میکنید میتواند H- را حذف کنید

این دستور چنین چیزی را بر می گرداند

Subschema

dn: cn=Subschema

.۰ NAME 'objectIdentifierMatch'۳۱.۵.۲ matchingRules: ( )۸۳.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

NAME 'distinguishedNameMatch'۱.۳۱.۵.۲ matchingRules: ( )
۲۱.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

NAME 'caseIgnoreMatch'۲.۳۱.۵.۲ matchingRules: (۵)۱.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'caseIgnoreOrderingMatch'۳.۳۱.۵.۲ matchingRules: (۵)
۱.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'caseIgnoreSubstringsMatch'۴.۳۱.۵.۲ matchingRules: ( )
۸.۵ ۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

.۵ NAME 'caseExactMatch'۳۱.۵.۲ matchingRules: (۵)۱.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'caseExactOrderingMatch'۶.۳۱.۵.۲ matchingRules: (۵)
۱.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

.۷ NAME 'caseExactSubstringsMatch'۳۱.۵.۲ matchingRules: ( )
۸.۵ ۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'numericStringMatch'۸.۳۱.۵.۲ matchingRules: ( )۶۳.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

۰ NAME 'numericStringSubstringsMatch'۱.۳۱.۵.۲ matchingRules: ( )
۸.۵ ۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'booleanMatch'۳۱.۳۱.۵.۲ matchingRules: (۰۷)۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'integerMatch'۴۱.۳۱.۵.۲ matchingRules: (۷)۲.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

۵ NAME 'integerOrderingMatch'۱.۳۱.۵.۲ matchingRules: (۷)
۲.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'bitStringMatch'۶۱.۳۱.۵.۲ matchingRules: ( )۶.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

۷ NAME 'octetStringMatch'۱.۳۱.۵.۲ matchingRules: (۰)۴.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'octetStringOrderingMatch'۸۱.۳۱.۵.۲ matchingRules: (۰)
۴.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

matchingRules:۰ NAME 'telephoneNumberMatch'۲.۳۱.۵.۲(

.۵۰)۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'telephoneNumberSubstringsMatch'۱۲.۳۱.۵.۲ matchingRules: ( )
۸.۵ ۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'uniqueMemberMatch'۳۲.۳۱.۵.۲ matchingRules: ( )۴۳.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

۷ NAME 'generalizedTimeMatch'۲.۳۱.۵.۲ matchingRules: ( )
۴۲.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'generalizedTimeOrderingMatch'۸۲.۳۱.۵.۲ matchingRules: ( )
۴۲.۱.۱۲۱ ۵.۱۱.۶۶۱۴.۱.۱۴.۱.۶.۳.۱ SYNTAX

NAME 'integerFirstComponentMatch'۹۲.۳۱.۵.۲ matchingRules: (۷)

۲.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

۰ NAME 'objectIdentifierFirstComponentMatch'۳.۳۱.۵.۲ matchingRules: ( )
۸۳.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

NAME 'certificateExactMatch'۴۳.۳۱.۵.۲ matchingRules: ( )۱ ۰.۷.۱۸۴۴۳۳.۱.۰.۶۲۸.۲.۱ SYNTAX

NAME 'caseExactIA۵Match'۱.۴۱۱.۹ ۰ ۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ matchingRules: ( )
۶۲.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

NAME 'caseIgnoreIA۵Match'۲.۴۱۱.۹ ۰ ۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ matchingRules: ( )
۶۲.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

NAME 'caseIgnoreIA۵SubstringsMatch'۳.۴۱۱.۹ ۰ ۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ matchingRules: ( )
۶۲.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

NAME 'caseExactIA۵SubstringsMatch'۱.۲.۱.۳ ۰ ۲۴.۱.۴.۱.۶.۳.۱ matchingRules: ( )
۶۲.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

NAME 'integerBitAndMatch'۳ ۰ ۸.۴.۱.۶ ۵۵ ۳۱۱ ۰.۴۸.۲.۱ matchingRules: (۷)
۲.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

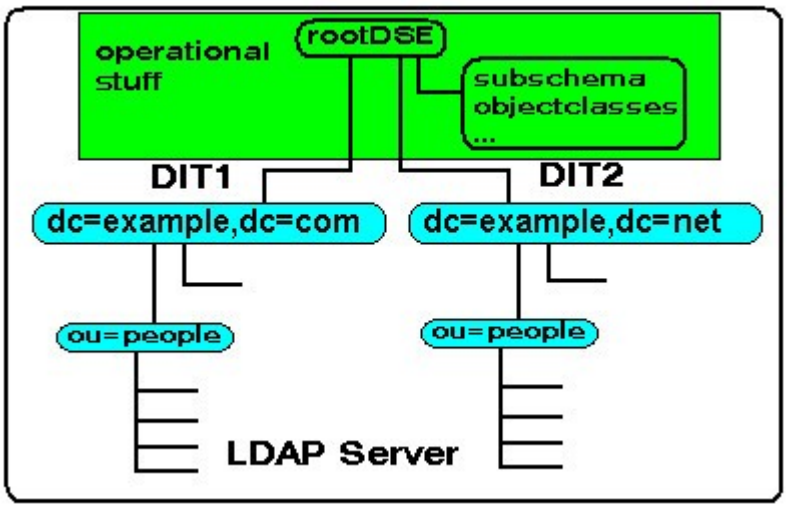
NAME 'integerBitOrMatch'۴ ۰ ۸.۴.۱.۶ ۵۵ ۳۱۱ ۰.۴۸.۲.۱ matchingRules: (۷)
۲.۱.۱۲۱ ۵.۱۱.۶۶۴۱.۱.۴.۱.۶.۳.۱ SYNTAX

شما میتوانید تعریف انگلیسی این OID ها را در این سایت بیابید.

آبجکت ها و اتریبیوت های عملیاتی ال دپ

تعدادی آبجکت و اتریبیوت در ال دپ به صورت درونی وجود دارند که تعریف میکنند چگونه ال دپ کار میکند و ما آنها را عملیاتی می نامیم.

این آبجکت ها و اتریبیوت ها اغلب در rootDSE قرار دارند و در حالت عادی دیده نمی شوند.
دیاگرام زیر نحوه ارتباط روت دی اس ای و آبجکت هایش با مدخل ها را نشان می دهد.



با یک ال دپ بروزر میتوانید این آبجکت ها و اتریبیوت ها را ببینید.

با دستور زیر میتوانید این عملیات ها را ببینید

ldapsearch -H ldap://ldap.mydomain.com -x -s base -b "" +

این دستور چیزی شبیه خروجی زیر را برمیگرداند

dn:

structuralObjectClass: OpenLDAProotDSE configContext: cn=config

namingContexts: dc=example,dc=com namingContexts: dc=example,dc=net
monitorContext: cn=Monitor

supportedControl:۱.۳.۶.۱.۴.۱.۲۱۴.۰ (Contentsync RFC ۱.۱.۹.۱.۳ ۴ ۵ ۳۰)

supportedControl:۱.۲.۸۴۰.۱۱۳۷۳۰.۳.۴.۲ ۰.۳ (ProxiedAuthv ۸۱.۴.۳ RFC ۲ ۳۴۷۰)

supportedControl:۱.۳.۶.۱.۴.۱.۲۱۴.۰ (ManageDSAIT RFC ۲.۴.۳ ۰.۳ ۷ ۳۱۱.۱ ۰.۴۸.۶۱.۲ ۳) ۷ ۶۳ (SubEntries
RFC ۱ ۰.۱.۱.۳ ۰ ۲۱۴.۱.۴.۱.۶.۳.۱:supportedControl
۶۹۶۲) (pagedResults RFC ۹۱۳.۴.۱.۶ ۵۵ ۳۱۱ ۰.۴۸.۲.۱:supportedControl ۶ ۷ ۸۳ (MatchedValues
RFC ۳.۲ ۰.۱۸۴۴۳۳.۱.۰.۶۲۸.۲.۱:supportedControl ۷)۲ ۵ ۴ (Post Read
RFC ۲.۳۱.۱.۱.۶.۳.۱:supportedControl
۷))۲ ۵ ۴ (Pre-Read RFC ۱.۳۱.۱.۱.۶.۳.۱:supportedControl
۸۲) ۵ ۴ (Assertion RFC ۲۱.۱.۱.۶.۳.۱:supportedControl

۸۸۰ ۳) (ModifyPassword RFC ۱.۱۱.۱.۳ ۰ ۲۱۴.۱.۴.۱.۶.۳.۱:supportedExtension
۲۳) ۵ ۴ (WhoAmI RFC ۳.۱۱.۱.۳ ۰ ۲۱۴.۱.۴.۱.۶.۳.۱:supportedExtension
۹۰ ۹۳) (Cancel RFC ۸.۱.۱.۶.۳.۱:supportedExtension
۵)۲ ۵ ۴ (Modify-Increment RFC ۴۱.۱.۱.۶.۳.۱:supportedFeatures
۴ ۷ ۶۳) (OperationalAttrs RFC ۱.۵.۱.۳ ۰ ۲۱۴.۱.۴.۱.۶.۳.۱:supportedFeatures
۹۲ ۵ ۴) (ObjectClassAttrs RFC ۲.۵.۱.۳ ۰ ۲۱۴.۱.۴.۱.۶.۳.۱:supportedFeatures
۶۲ ۵ ۴) (TrueFalse RFC ۳.۵.۱.۳ ۰ ۲۱۴.۱.۴.۱.۶.۳.۱:supportedFeatures

supportedFeatures:۱.۳.۶.۱.۴.۱.۲۱۴ ۰ ۴.۵.۱.۳ RFC LanguageTag) ۶۶۸۳)
supportedFeatures:۱.۳.۶.۱.۴.۱.۲۱۴ ۰ ۱.۳ RFC LanguageRange) ۶۶۸۳.۵.۵)
supportedLDAPVersion:۳

supportedSASLMechanisms: NTLM

supportedSASLMechanisms: GSSAPI

supportedSASLMechanisms: DIGEST-MD۵
supportedSASLMechanisms: CRAM-MD۵
entryDN:

subschemaSubentry: cn=Subschema

توضیح هر کدام از این عملیات ها در این سایت آمده است.

بخش ۲)تنظیمات اولیه ال دپ

دایرکتوری ساده

با مثالی ساده و امن نشده شروع میکنیم

وقتی با چیزی به پیچیدگی ال دپ کار میکنیم تقریبا ممکن است ۶ میلیون اشکال به وجود آید.
ما برای کاهش تعداد ۳ میلیون مشکل امنیت را نادیده گرفته ایم.

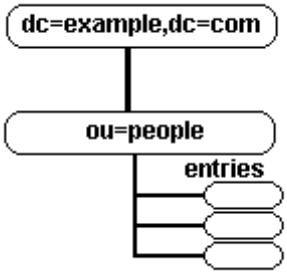
بنابراین داده های مهم را در این برنامه قرار ندهید. می توانید به امنیت به شکل یک افزونه نگاه کنید.
شما هر زمان خواستید میتوانید آن را به برنامه بیفزایید بدون اینکه در داده اصلی تغییری ایجاد شود.

طرای یک دیت

این یک مرحله مهم است و شما میتوانید تمام عمر خود را صرف طرامی دیت خود کنید .حتی کتابهایی نیز در این مورد یافت می شوند.

چند نمونه قابل قبول در این رابطه اینجا آورده شده اند

ما از فرمت ۷۷ ۳۲ RFC کلسیک تبعیت میکنیم و از dc فرمت برای روت دیت خود استفاده میکنیم ساختار را در people توسعه می دهیم.میخواهیم یک دفترچه تلفن بنویسیم و از اتریبیوت ou برای دپارتمان افراد استفاده میکنیم. ساختار ما شبیه شکل زیر می شود.



انتخاب یک آبجکت کلاس ساختاری

بسیاری از گله ها و مشکلات در انتخاب یک آبجکت کلاس اولیه است .و تبدیل به یک رسم در ال دپ شده است. زودتر شروع به تمرین کنیم.

برای بسیاری از استفاده های معمول و ساختار های خیلی زیاد معمول inetorgperson انتخاب مناسبی است. به سبب ساختار درختی بزرگ آن و تعدد اتریبیوت ها آن. اگر چیزی را فراموش کردیم بعدا میتوانیم آنرا اضافه کنیم. کاری که در مرحله ۴ انجام داده ایم. خوب تصمیم گرفتیم و موضوع فعل تمام شده است.

فایل slapd.conf

این یک فایل مثال slapd.conf است و اجازه دهید از بک اند(Oracle Berkeley DataBase (BDB استفاده کنیم). ال دب هم اکنون این را پیشنهاد می دهد.

```
#
- SIMPLE DIRECTORY ##########١###### SAMPLE
#
# NOTES: inetorgperson picks up attributes and objectclasses #       from all three schemas
#
# NB: RH Linux schemas in /etc/openldap
#
include/usr/local/etc/openldap/schema/core.schema
include/usr/local/etc/openldap/schema/cosine.schema
include/usr/local/etc/openldap/schema/inetorgperson.schema
# NO SECURITY - no access clause
# defaults to anonymous access for read
```

only rootdn can write

NO REFERRALS

DON'T bother with ARGS file unless you feel strongly # slapd scripts stop scripts need this to work
pidfile /var/run/slapd.pid

enable a lot of logging – we might need it # but generates huge logs

loglevel 1-

MODULELOAD definitions

2.4# not required (comment out) before version moduleload back_bdb.la

NO TLS-enabled connections

backend definition not required

############

bdb database definitions
#

replace example and com below with a suitable domain
#

If you don't have a domain you can leave it since example.com
is reserved for experimentation or change them to my and inc
#

############

database bdb

suffix "dc=example, dc=com"

 # root or superuser

rootdn "cn=jimbob, dc=example, dc=com" rootpw dirtysecret

The database directory MUST exist prior to running slapd AND # change path as necessary

directory/var/db/openldap/example-com

Indices to maintain for this directory
unique id so equality match only

index uid eq

allows general searching on commonname, givenname and email index cn,gn,mail eq,sub

allows multiple variants on surname searching index sn eq,sub

sub above includes subintial,subany,subfinal # optimise department searches

index ou eq

if searches will include objectClass uncomment following # index objectClass eq

shows use of default index parameter index default eq,sub

indices missing – uses default eq,sub index telephonenumber

other database parameters

read more in slapd.conf reference section ۱۰۰۰۰ cachesize

۵۱۸۲۱ checkpoint

نکات:

۱)ما میتوانیم تنظیمات ویژه دیتابیس را تغییر دهیم. تنظیمات cachesize و checkpoint با دلیل انجام شده اند و باعث میشوند بسیاری خطاهای نوشتن در دیتابیس از جلوگیری کند. اگر برای شما عملکرد و پرفرمنس مهم است میتوانیم با یادگیری مفاهیم تنظیمات BDB آنها را بر اساس نیاز خود تغییر دهید.

۲) امنیت با استفاده از دایرکتوری های access تعریف می شود. در اینجا ما اجازه میدهیم که کاربر مهمان داده ها را بخواند). احراز هویت لزومی ندارد (بدون اجازه نوشتن).

همانطور که در فایل کانفیگ rootdn و rootpw آمده است ما میتوانیم با استفاده از این dn تعریف شده در دایرکتوری بنویسیم و با bind در دایکتوری مدخل ایجاد کنیم.

۳) هیچ دستور بازگشتی وجود ندارد. و لزومی هم به این کار نیست.
۴) ایندکس انتخاب شده برای بهینه کردن جستجو انجام شده است. شما میتوانید بدون ایندکس هم جستجو کنید، منتها زمان بیشتری نیاز دارد.

فایل LDIF

ال دیف زیر یک ساختار دیت را ایجاد میکند و در یک فرد را به pepole اضافه می کند.
ال دیف با استفاده از ابزار ldapadd اضافه شده است). وقتی که slaps در حال اجرا است(
قبل از ساخت ال دیف ما باید معلوم که چه داده هایی الزامی هستند. با نگاهی سریع به ساختار آبجکت کلاس inetorgperson میفهمیم که تنها دو اتریبیوت cn و sn الزامی هستند. ال دیف زیر یک ساختار اولیه بر روی دیت

نصب میکند و بعد یک فرد را به ساختار اضافه می کند.

DEFINE DIT ROOT/BASE/SUFFIX #### ۷۷ format ۳۲## uses RFC

replace example and com as necessary below ## or for experimentation leave as is

dcObject is an AUXILLIARY objectclass and MUST

have a STRUCTURAL objectclass (organization in this case)
this is an ENTRY sequence and is preceded by a BLANK line

dn: dc=example,dc=com

dc: example

description: My wonderful company as much text as you want to place K continuation
data for the line above must ۲۳ in this line up to
have <CR> or <CR><LF> i.e. ENTER works

on both Windows and *nix system - new line MUST begin with ONE SPACE

objectClass:dcObject

objectClass: organization o: Example, Inc.

FIRST Level hierarchy - people

uses mixed upper and lower case for objectclass

this is an ENTRY sequence and is preceded by a BLANK line

dn: ou=people, dc=example,dc=com

ou: people

description: All people in organisation objectclass: organizationalunit

SECOND Level hierarchy

ADD a single entry under FIRST (people) level

this is an ENTRY sequence and is preceded by a BLANK line # the ou: Human Resources is the department name

dn: cn=Robert Smith,ou=people,dc=example,dc=com objectclass: inetOrgPerson

cn: Robert Smith

cn: Robert J Smith
cn: bob smith
sn: smith

uid: rjsmith

userpassword: rJsmitH

۳۲۱ carlicense: HISCAR

۲۲۲۲-۱۱۱ homephone: ۵۵۵-

mail: r.smith@example.com

mail: rsmith@example.com

mail: bob.smith@example.com description: swell guy

ou: Human Resources

نکات:

۱)کامنت ها با#شروع می شوند.

۲) حداقل یک خط فاصله باید قبل از شروع هر dn باشد.

۳) توضیحات چند خطی در صورتی فهمیده میشوند که حتما با خط جدید(new line) تمام شوند و در خط بعد با یک کاراکتر فاصله شروع شوند

۴) در بسیاری از تعاریف شما objectclass: top را می بینید. به دلیل اینکه باید تمام آبجکت کلاسها تعریف شده باشند. اما اپن ال دپ ۲ اجباری در این کار قرار نداده است.

۵) فاصله بعد از هر : اجباری است

در بسیاری از مستندات شما میبینید که مدخل در ال دیف برای rootdn (کاربر مدیر) تعریف شده است. در مثال بالا

cn=jimbob,dc=example,dc=com

استفاده از rootpw به سبب دسترسی خارجی که ایجاد می کند بسیار خطرناک است. یک بحث کامل در اینجا درباره آن انجام شده است.

لود کردن یک ال دیف

باید از قبل سرور ال دپ راه اندازی شده باشد.

[redhat] /etc/rc.d/init.d/ldap start

[bsd] /usr/local/etc/openldap/slapd.sh start

confirm slapd is running ps ax | grep slapd

(you should see the slapd process entry if it has been started successfully)

ما ال دیفی را به بالا را در createdit.ldif در پوشه tmp ذخیره کردیم

ldapadd -H ldap://ldaphost.example.com -x -D "cn=jimbob,dc=example,dc=com" -f /tmp/createdit.ldif -w dirtysecret

اگر از سرور لوکال استفاده میکنید میتوانید H- را حذف کنید.

x-نشان میدهد که ما از SASL security استفاده نمیکنیم و از اپن ال دپ ۲ به بعد ضروری است

D-برای بیند (bind) لازم است و به دلیل اینکه ما از rootdn استفاده کرده ایم و دسترسی دیگری تعریف نکرده ایم .کاربر مدیر میتواند هر کاری انجام دهد.

بعد از w- رمز عبور می آید .امن نیست اما برای کل دایرکتوری تعریف شده است.

افزودن یک مدخل با استفاده از یک ال دیف

ال دیف زیر نشان میدهد که چگونه میتوان یک مدخل جدید اضافه کرد.

۱:version

ADD a single entry to people level

dn: cn=John Smith,ou=people,dc=example,dc=com objectclass: inetOrgPerson

cn: John Smith

cn: John J Smith
sn: Smith

uid: jsmith

userpassword: jSmitH

۴۲۱ carlicense: HISCAR

۳۲۲۲-۱۱۱ homephone: ۵۵۵-

mail: j.smith@example.com

mail: jsmith@example.com

mail: john.smith@example.com
ou: Sales

ADD another single entry to people level

dn: cn=Sheri Smith,ou=people,dc=example,dc=com objectclass: inetOrgPerson

cn: Sheri Smith

sn:smith

uid: ssmith

userpassword: sSmitH

۵ ۲۱ carlicense: HERCAR

۵ ۲۲۲-۱۱۱ homephone: ۵۵۵-

mail: s.smith@example.com

mail: ssmith@example.com

mail: sheri.smith@example.com
ou: IT

بعد از ذخیره کردن آن در یک فایل آن را با دستور زیر اضافه میکنیم

ldapadd -H ldap://ldaphost.example.com -x -D "cn=jimbob,dc=example,dc=com" -f /tmp/addentry.ldif -w dirtysecret

تغییر یک مدخل

ال دیف زیر نشان میدهد که چگونه میتوان یک مدخل را تغییر داد

استفاده از یک ال دپ بروزر راحت تر است اما برای حجم زیاد تغییرات دستورات کامند سریعتر است.

۱:version

MODIFY the Robert Smith entry

dn: cn=Robert Smith,ou=people,dc=example,dc=com changetype: modify

add: telephonenumber

۲۱۲۱ telephonenumber: ۵۵۵-۵۵۵-۲۱۲:telephonenumber

-

replace: uid
uid: rjosmith

-

replace: mail

mail:robert.smith@example.com

mail: bob.smith@example.com

-

adds using URL format add: jpegphoto

jpegphoto: < file://path/to/jpeg/file.jpg

-

delete: description

نکات:

با دستور changetype: modify ما به ال دپ میگوییم که به چه مودی برود. دستوری نیز به نام :changetype delete داریم برای با دستور مود حذف با add: attr ما به ال دپ میگوییم که میخواهیم یک اتریبیوت اضافه کنیم در اینجا تلفن و آدرس اضافه کردیم

دستور file://path/to/jpeg/file.jpg >:jpegphoto میگوید که محتویات فایل را بخوان .دستور آن با رعایت فاصله دقیقا باید رعایت شود.

۴) delete: description برای حذف description استفاده می شود.

خط مشی امنیت

حال ما چند مورد امنیتی ساده به فایل کانفیگ ال دپ خود اضافه میکنیم

تعریف سطحهای دسترسی بر اساس خط مشی سازمانی تعریف می شود(Access Control Policy)

چند مورد از تعاریف این خط مشی ها میتواند بصورت زیر باشد.

۱) صاحب مدخل میتواند تمام اتریبیوت های بروز رسانی شده شامل رمزهای عبور را ببیند.

۲) منابع انسانی میتواند تمام مدخل ها را بروز رسانی کند اما نمیتواند رمز های عبور را تغییر دهد.

۳) مدخل های carlicence,homepostaddress,homephoneتوسط هیچکس غیر منابع انسانی و صاحب مدخل قابل خواندن نیست

۴) تمام کاربران باید تأیید هویت شوند کاربر میهمان هیچ اجازه ای ندارد

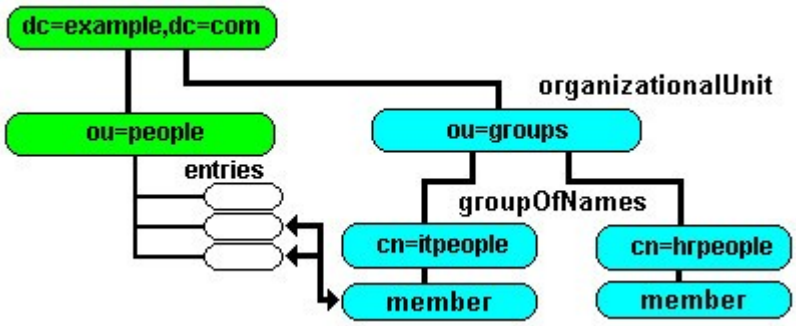
۵) مسئول آی تی میتواند تمام رمز های عبور را بروزرسانی کند.

هر ایده ای که در این رابطه داشته باشید باید میتوانید با اکسس کنترل به اجرا در آورید.

اولین کاری که باید انجام دهید این است که دو گروه ایجاد کنید

یکی به نام hrpeople و دیگری به نام itpeople و سطح دسترسی های گروه را تعریف کنید.

شما میتوانید این گروهها را با کلید واژه groups در دیت روت تعریف کنید.



ال دیف زیر نشان میدهد چطور میتوانید گروهها را اضافه کنید

۱:version

create FIRST Level groups branch

dn: ou=groups,dc=example,dc=com
objectclass:organizationalunit
ou: groups

description: generic groups branch

create the itpeople entry under groups

dn: cn=itpeople,ou=groups,dc=example,dc=com objectclass: groupofnames

cn: itpeople

description: IT security group

member: cn=William Smith,ou=people,dc=example,dc=com

create the hrpeople entry under groups

dn: cn=hrpeople,ou=groups,dc=example,dc=com objectclass: groupofnames

cn: hrpeople

description: Human Resources group

member: cn=Robert Smith,ou=people,dc=example,dc=com

نکات:

۱) ما از آبجکت کلاس برای تعریف groupOfNames استفاده کرده ایم

۲) member: dn کاربران را با استفاده از dn آنها اضافه می کند.

ldapadd -H ldap://ldaphost.example.com -x -D "cn=jimbob,dc=example,dc=com" -f /tmp/addgroups.ldif -w dirtysecret

حال ما باید خط مشی دسترسی خود را در slapd.conf تعریف کنیم

کد های زیر فایل کانفیگ ما را با استفاده از کد های سطح دسترسی اضافه شده نشان می دهد.

#

- DIRECTORY with ACL ##########۲###### SAMPLE

#

NOTES: inetorgperson picks up attributes and objectclasses # from all three schemas

#

NB: RH Linux schemas in /etc/openldap

#

include/usr/local/etc/openldap/schema/core.schema

include/usr/local/etc/openldap/schema/cosine.schema

include/usr/local/etc/openldap/schema/inetorgperson.schema

NO REFERRALS

DON'T bother with ARGS file

```
# pidfle allows scripts for stopping slapd to work pidfile /var/run/slapd.pid

# enable a lot of logging - we might need it

loglevel 1-

# NO dynamic backend modules

# NO TLS-enabled connections

#########################################

# bdb database definitions
#

# replace example and com below with a suitable domain
#

# If you don't have a domain you can leave it since example.com
# is reserved for experimentation or change them to My and inc

#########################################

database bdb

suffix "dc=example, dc=com"
1# ACL

access to attrs=userpassword
by self      write

by anonymous  auth

by group.exact="cn=itpeople,ou=groups,dc=example,dc=com"
write

by *none

9# ACL

access to attrs=carlicense,homepostaladdress,homephone
by self      write

by group.exact="cn=hrpeople,ou=groups,dc=example,dc=com"
write

by *none

B# ACL

access to *
```

by self write

by group.exact="cn=hrpeople,ou=groups,dc=example,dc=com"
write

by users read

by *none

root or superuser

rootdn "cn=jimbob, dc=example, dc=com" rootpw dirtysecret

The database directory MUST exist prior to running slapd AND # change path as ncessary

directory/var/db/openldap/example-com

Indices to maintain for this directory
required if searches will use
unique id so equality match only
index uid eq

allows general searching on commonname, givenname and email index cn,gn,mail eq,sub

allows multiple variants on surname searching index sn eq,sub

sub above includes subintial,subany,subfinal # optimise department searches

index ou eq

if searches will include objectClass uncomment following # index objectClass eq

shows use of default index parameter index default eq,sub

indices missing - uses default eq,sub index telephonenumber

other database parameters

read more in slapd.conf reference section ۰۰۰۰۱ cachesize

۵۱۸۲۱ checkpoint

حال ال دپ را ریستارت میکنیم

stop and start OpenLDAP (slapd)

on Linux/Redhat

/etc/rc.d/init.d/ldap restart

on BSD

[bsd] /usr/local/etc/rc.d/slapd.sh stop
then

[bsd] /usr/local/etc/rc.d/slapd.sh start

confirm slapd is running

ps ax | grep slapd

.....

LDIF File Format ۱.۲.۸

LDIF Terminologyand Line Types ۱.۱.۲.۸

LDIF Sample ۲.۱.۲.۸

LDIF Directives ۲.۲.۸

add Directive ۱.۲.۲.۸

attributename Directives ۲.۲.۲.۸

changetype Directives ۳.۲.۲.۸

control Directives ۴.۲.۲.۸

delete Directives ۲.۲.۸ ۵.

deleteoldrdn Directives ۶.۲.۲.۸

dn Directives ۲.۲.۸ ۷.

newrdn Directives ۸.۲.۲.۸

newsuperior Directives ۹.۲.۲.۸

objectclass Directives ۱.۲.۲.۸ ۰

replace Directives ۱۱.۲.۲.۸

version Directives ۲۱.۲.۲.۸

LDIF Handling Binary (including Passwords)۳.۸

LDIF Importing Files ۴.۸

LDIF Samples ۸ ۵.

DSML ۶.۸

OpenLDAP HowTos ۱۱ .

Configuring Multiple DITs in OpenLDAP

Configuring Referrals in OpenLDAP

Configuring Referral chaining in OpenLDAP

Configuring slurpd style replication in OpenLDAP

Configuring syncrepl style replication in OpenLDAP

Configuring delta synchronization (syncrepl) in OpenLDAP Configuring and using cn=config in OpenLDAP

Notes about running/initialising OpenLDAP

Notes about overlays in OpenLDAP (or when is an overlay an overlay) OpenLDAP (converting to OLC (cn=config

(Using OLC (cn=config

Configuring Groups of Users in OpenLDAP

OpenLDAP Troubl ۲۱ e Shooting & Errors .

OpenLDAP Performance ۳۱ .

LDAP Tools ۴۱ .

LDAP Security ۱ .۵

OpenLDAP Security Overview ۱ ۵.۱

OpenLDAP TLS/SSL Configuration ۴ ۵.۱

Appendix A: LDAP Notes and Explanations Appendix B: LDAP Resources

Appendix C: LDAP RFCs and Documentation Appendix D: LDAP Glossary

Appendix E: LDAP Schemas, objectClasses and Attributes

.....

ابزار های شل ال دپ

ldapadd – add LDIF entries to an LDAP directory ldapauth – add LDIF entries to an LDAP directory ldapdelete – delete LDAP entries

ldapmodify – modify existing LDAP entries
ldapmodrdn – modify an LDAP entry's DN
ldappasswd – modify an entry's password
ldapsearch – search LDAP entries

ldapwhoami – perform an LDAP Who Am I operation of a server

slapacl – verify access to attributes by inspecting the configuraion of a DIT

slapadd – add LDAP entries to a database – STOP SLAPD FIRST slapauth – verify SASL data against a DIT

slapcat – export an LDIF from an LDAP database – STOP SLAPD FIRST slapdn – verify a DN against a DIT configuration

slapindex – re-index an LDAP database – STOP SLAPD FIRST slappasswd – generate password

slaptest – verify a slapd.conf file or a cn=config directory (slapd.d)

برنامه های ال دپ بروزر

phpLDAPadmin

LDAP Browser/Editor
JXPlore

FusionDirectory

LDAPExplorerTool

Apache LDAP Studio

openLDAP

انواع مختلف داده در openldap

Strings

Numbers (Integer)
Time

Telephone Numbers
Boolean

Binary

Distinguished Name Bit Strings

LDAP Supported Syntaxes

کلید واژه ها:

LDAP,Distinguished Names (DNs), Schemas, object classes,attribute, slapd,openLDAP,LDIF

برنامه های معادل openLDAP

۳۸۹ Directory Server
OpenDS

(ApacheDS (Apache Directory

مقایسه بین نرم افزار های سرویس دهنده LDAP

http://www.zytrax.com/books/ldap/implementations.html

واژه نامه:

http://www.zytrax.com/books/ldap/apd/

مراجع:

http://www.python-ldap.org/docs.shtml

http://www.openldap.org/doc/

http://www.zytrax.com/books/ldap/ch۱/