
بسم الله الرحمن الرحیم

پیکر بندی سرورهای شبکه

دانشکده فنی حرفه ای شهرستان صومعه سرا

مدیریت گروه شبکه های کامپیوتری

به کوشش مهندس سُها محمدی ریک

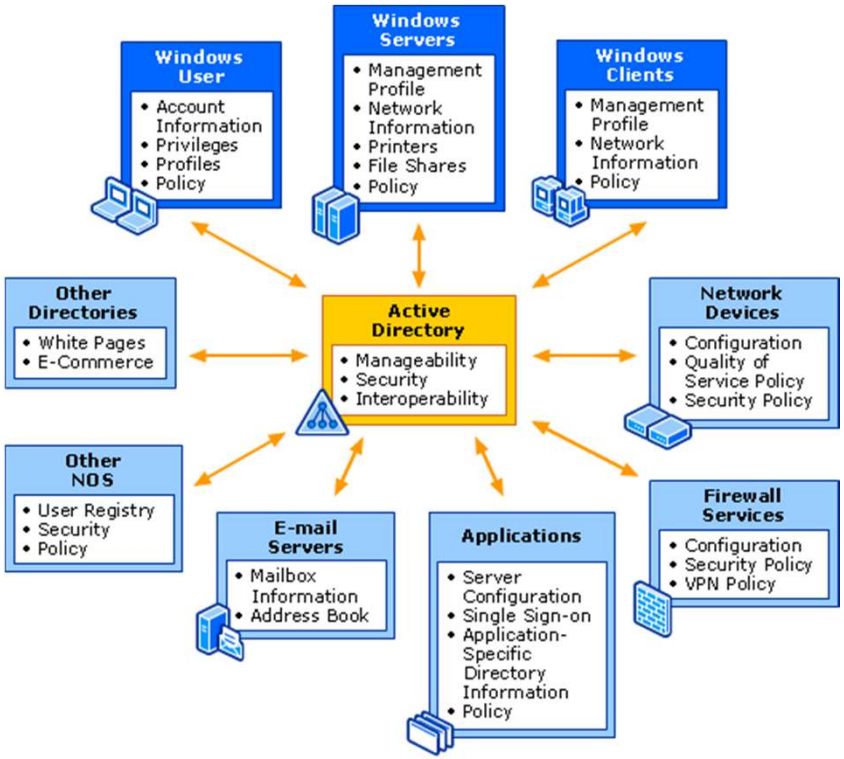
https://sohamohammadi.ir

mail@sohamohammadi.ir

۰۹۱۲۴۲۵۷۸۸۳

پیکر بندی سرورهای شبکه

Active Directory



سرفصل درس پیکر بندی سرورهای شبکه

۱- مفاهیم پایه مدیریت شبکه Client/Server

- User account
- Domain
- Policy

۲- معرفی ابزار Active directory و کاربرد آن در طراحی Infrastructure شبکه

۳- مفهوم forest و domain و طراحی آن به کمک Active directory

۴- مفهوم DNS ، Name Resolution Strategy و طراحی DNS Namespace

۵- معرفی انواع Policy و طراحی یک Schema management policy

۶- طراحی Site Infrastructure و معـرفی Domain Controllers ، Global Catalog Servers و Single Operation Master

۹- طراحی Network Connectivity Plan ، Connection Type

شامـل:

Evaluation، Internet Connectivity ، Connectivity Infrastructure

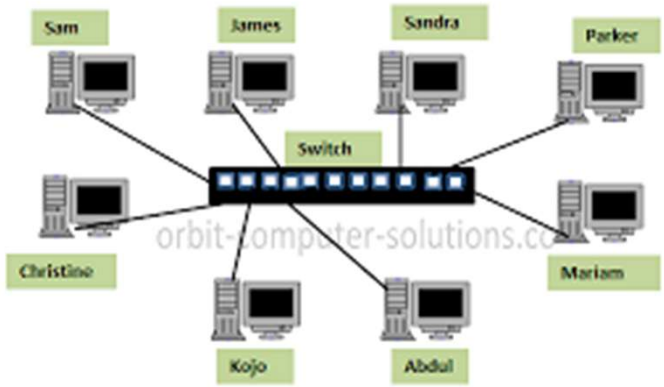
۱۰- توسعه و انتقال شبکه (Migration Plan, Trust strategy, Replication strategy,)

۱۱- طراحی زیرساختار دسترسی به شبکه شامل Remote Access, VLAN, Wireless

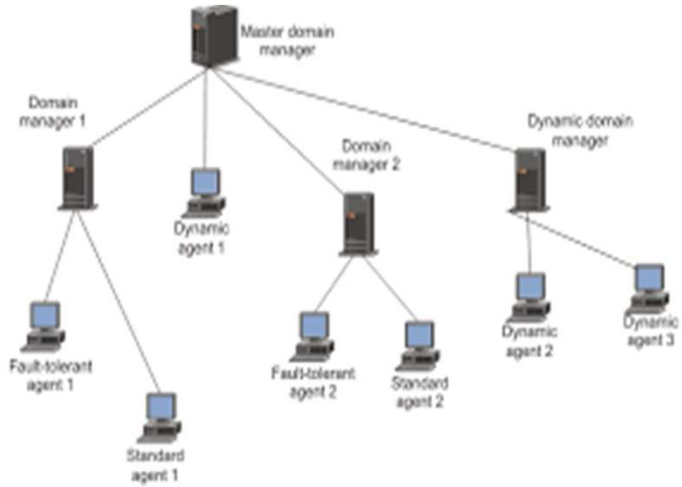
۱۲- مباحث مربوط به امن سازی شبکه های مبتنی بر Active Directory

در شبکه دو مدل سرویس دهی وجود دارد:

- نظیر به نظیرWorkgroup



- مبتنی بر سرور Domain



سرویس دهی نظیر به نظیر

در مدل سرویس دهی نظیر به نظیر (Workgroup) ، کاربری که با استفاده از یک حساب کاربری می تواند به یک کامپیوتر وارد شود برای ورود به کامپیوتر های دیگر نیازمند حسابهای کاربری جداگانه ای می باشد. در این شبکه ها استفاده از حسابهای کاربری متعدد برای کاربران یکسان چندان جالب نمی باشد زیرا هر کاربر با تعدادی از حسابهای کاربری و رمز عبورهای متفاوت سروکار دارد.

در واقع شبکه Workgroup معمولاً برای سازمان هایی که دارای حدود ۱۰ کامپیوتر هستند مناسب است. چون در شبکه Workgroup مدیر مرکزی وجود ندارد و هر کاربر مدیر رایانه خودش می باشد.

در چنین مدلی اگر لازم باشد یک سیاست امنیتی یا مدیریتی برای رایانه ها یا کاربران شبکه تعیین می شود. این سیاست باید به صورت جداگانه در تک تک رایانه ها انجام گیرد که کار مشکلی است.

سرویس دهی مبتنی بر سرور

در مدل سرویس دهی مبتنی بر سرور (Domain) این امکان وجود دارد که بتوان تمامی رایانه ها یا کاربران و منابع موجود در شبکه را به صورت متمرکز مدیریت نمود. باید توجه داشت که شبکه Domain را فقط توسط یک سیستم عامل سروری که سرویس Active Directory بر روی آن نصب باشد می توان ایجاد نمود.

وقتی که شما می خواهید یک تماس تلفنی برقرار نمائید، شماره مورد نظر را از دفترچه تلفن پیدا می کنید، یا وقتی که در یک ساختمان اداری بزرگ به دنبال اتاق خاصی می گردید به راهنمای طبقات مراجعه می کنید و یا در کتابخانه در هنگام جستجوی یک کتاب خاص به مخزن کتابخانه مراجعه می کنید. دفترچه تلفن، دایرکتوری (Directory) محسوب می شوند.

دایرکتوری های شبکه، اطلاعاتی درباره منابع موجود روی شبکه همانند کاربران، رایانه ها، چاپگرهاو پوشه های به اشتراک گذاشته شده را نگهداری می نمایند.

اکتیو دایرکتوری یک پایگاه داده مرکزی در ویندوز سرور است که تمامی اطلاعات مربوط به منابع شبکه را در خود نگهداری می نماید. (بدون اکتیو دایرکتوری برای اینکه بفهمیم یک منبع در کجا قرار دارد باید بدانیم که کدام کامپیوتر اطلاعات مربوط به آن منبع را نگهداری می نماید).

مشخصات یک Active Directory خوب

۱- Centralization

اکتیو دایرکتوری خوب باید به اطلاعات مرکزیت بدهد یعنی برای دسترسی به یک سری اطلاعات نیاز به جستجو در مکان های مختلف نباشد بلکه یک منبع اطلاعاتی خوبی وجود داشته باشد و ما فقط به آن سر بزنیم.

۲- Scalability

وقتی امکانات شبکه زیاد می شود یا بعبارتی دیگر AD بزرگ می شود باید بتواند با آن گسترش کنار بیاید و سرعت آن کاهش پیدا نکند، همچنین مانند سابق پرسش ها را سریعاً جواب بدهد، مثلاً یک چاپگر خاص کجا قرار دارد.

۳- Standardization

بر اساس استانداردهای موجود دنیا تدوین شده باشد.

۴- Extensible

اکتیو دایرکتوری باید پذیرای افزایش قابلیت ها باشد. هر برنامه ای که به سیستم عامل اضافه می شود ممکن است بخواهد خودش به AD یک سری قابلیت ها را اضافه نماید و از آنها استفاده نماید در نتیجه AD باید پذیرای افزایش قابلیت ها باشد.

۵- Separation of physical network

اکتیو دایرکتوری باید بتواند ساختار فیزیکی شبکه را از ساختار منطقی آن جدا نماید و این هدف اصلی طراحی شبکه است و لازم نیست کاربر بداند امکانات فیزیکی شبکه در کجا قرار دارد تا از آنها استفاده نماید.

۶- Security

امنیت در ذخیره اطلاعات و امنیت در پرس و جوها و امنیت در استفاده از اطلاعات باید در AD وجود داشته باشد.

اجزای اکتیو دایرکتوری

اجزای اکتیو دایرکتوری به دو دسته تقسیم می شوند:

۱- ساختار منطقی

۲- ساختار فیزیکی

ساختار منطقی اکتیو دایرکتوری

در اکتیو دایرکتوری می توان منابع را به صورت یک ساختار منطقی سازمان دهی نمود.

سازمان دهی منطقی منابع سبب میشود که بدون توجه به محل فیزیکی منابع به جستجوی آنها بپردازیم و این امر باعث می شود که از به یادسپاری محل فیزیکی منابع بی نیاز شویم.

ساختارهای منطقی قابل لمس نیستند و به صورت فیزیکی وجود ندارند.

ساختار منطقی اکتیو دایرکتوری شامل موارد زیر است :

۱- اشیاء Object

۲- دامنه ها Domain

۳-واحد های سازمانی OU

۴- درخت Tree

۵- جنگل Forest

۱– اشیاء Object

در اکتیو دایرکتوری هر منبع یک Object محسوب می شود، مثلاً کامپیوتر، پرینتر، یوزر و یا هر منبع دیگری که در شبکه داریم یک Object محسوب می شود.

ویژگی ها Attribute

هر شیء یا Object توسط مجموعه ای از صفات و مقادیر مشخص می شود.

به عنوان مثال ویژگی های یک حساب کاربری می تواند شامل نام و نام خانوادگی و نام ورود برای آن کاربر باشد.

Object Class :

Object class الگو یا Template برای Object های موجود در شبکه است.

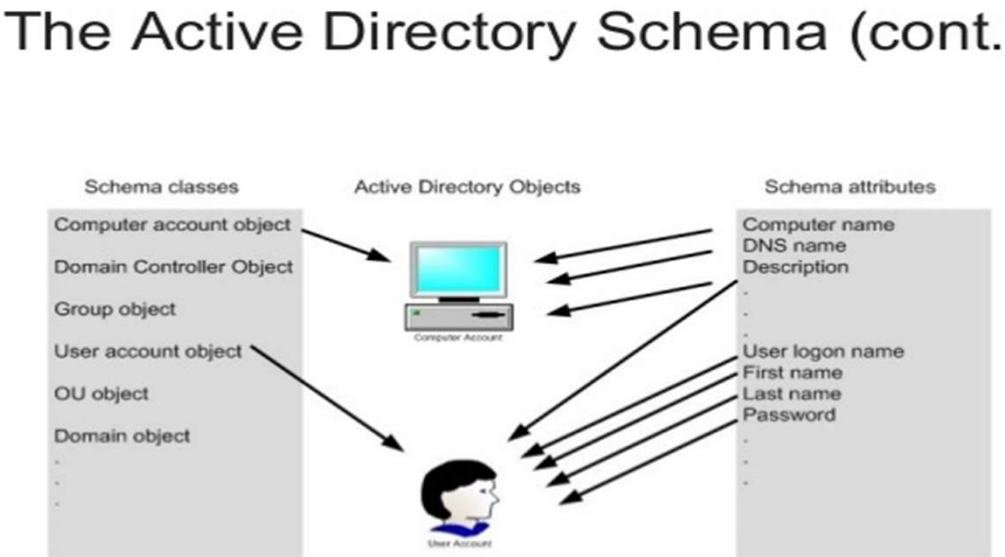
هر Object بر اساس Object class هایی که برای آن تعریف شده است ساخته می شود و برای ساختن یک Object حتماً باید یک الگو یا Template برای آن در Object Class وجود داشته باشد.

Object class مثل نقشه آماده ساختمان می ماند که از روی آن

می شود هزاران ساختمان (Object) را ایجاد نمود.

در اکتیو دایرکتوری به Object class و Attribute های موجود در آن اصطلاحاً Active directory schema گفته می شود.

در واقع Active directory schema جایی است که نقشه های مربوط به اشیاء (Object class) در آنها نگهداری میشود و در واقع وقتی یک Object قرار است ایجاد شود از روی نقشه های داخل Schema ساخته میشود.



۲– Domain دامنه:

دامنه (Domain) هسته ی اصلی ساختار AD است که می تواند میلیون ها شیء را در خود ذخیره نماید.

Domainیک حوزه مدیریتی در ویندوز سرور است. در واقع تعدادی شیء را در یک حوزه مدیریتی جمع می کنیم و به آن Domain می گوییم.

با استفاده از Domain می توان میلیون ها شیء را در یک Domain مدیریت نمود.

۳– واحد های سازمانی Organization unit (OU) :

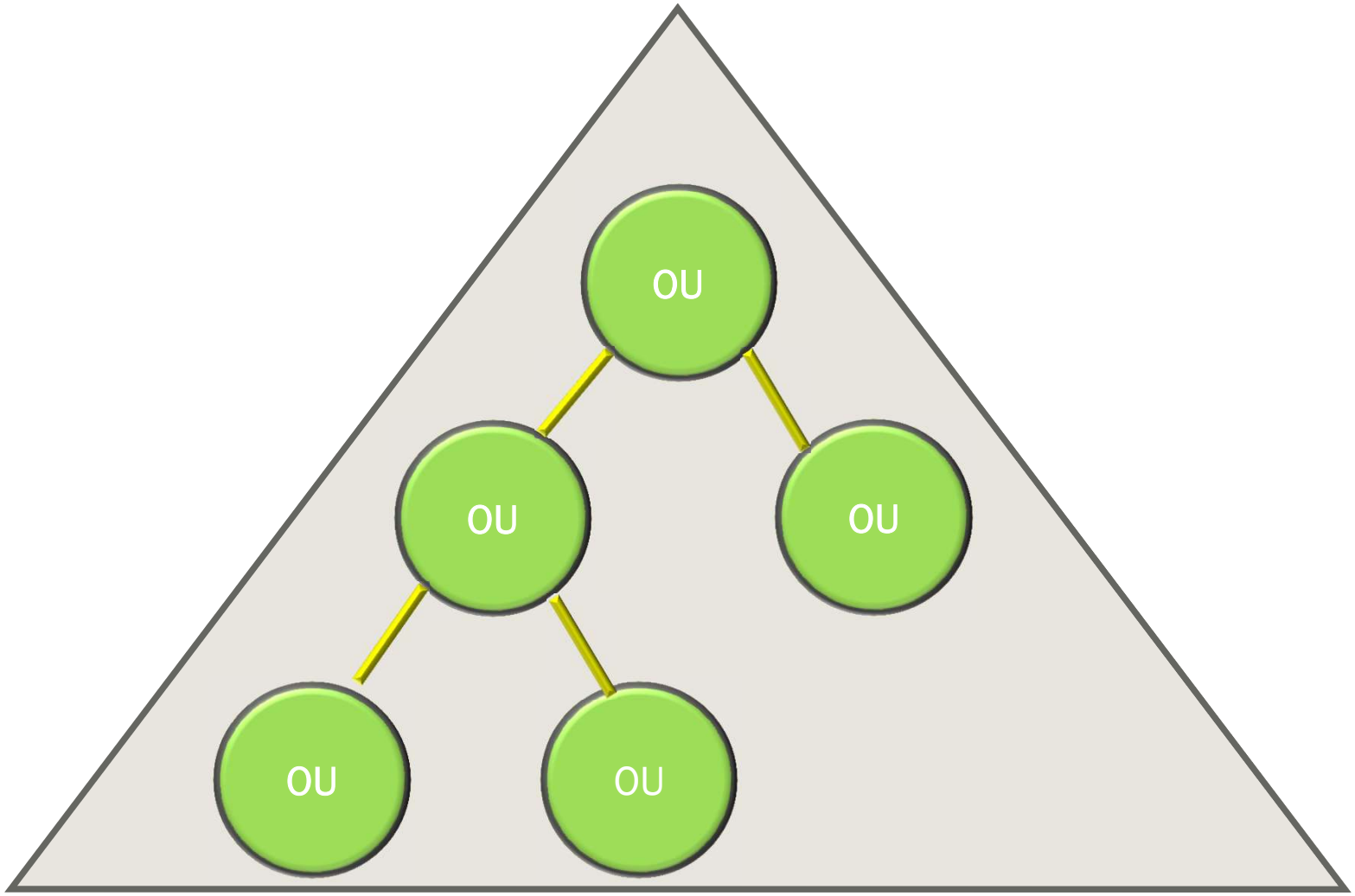
OU ها ظرف هایی هستند که شما در آنها می توانید Object های مختلف را قرار دهید ولی OU ها داخل دامین تعریف می شوند.

در واقع OU مانند یک ظرف است که اشیاء موجود در دامین را به گروه های کوچکتری تقسیم می نماید.

OU ها را می توان جداگانه مدیریت نمود و برای آنها قوانین و ضوابط جداگانه وضع نمود. برای مثال در یک شرکت ۳ بخش فروش ، مدیریت و تحقیق وجود دارد که هر کدام از آنها را در درون یک OU قرار می دهیم.

می توانیم برای هر OU یک مدیر تعریف نمائیم که تنها اجازه دسترسی به Object های درون همان OU را خواهد داشت.

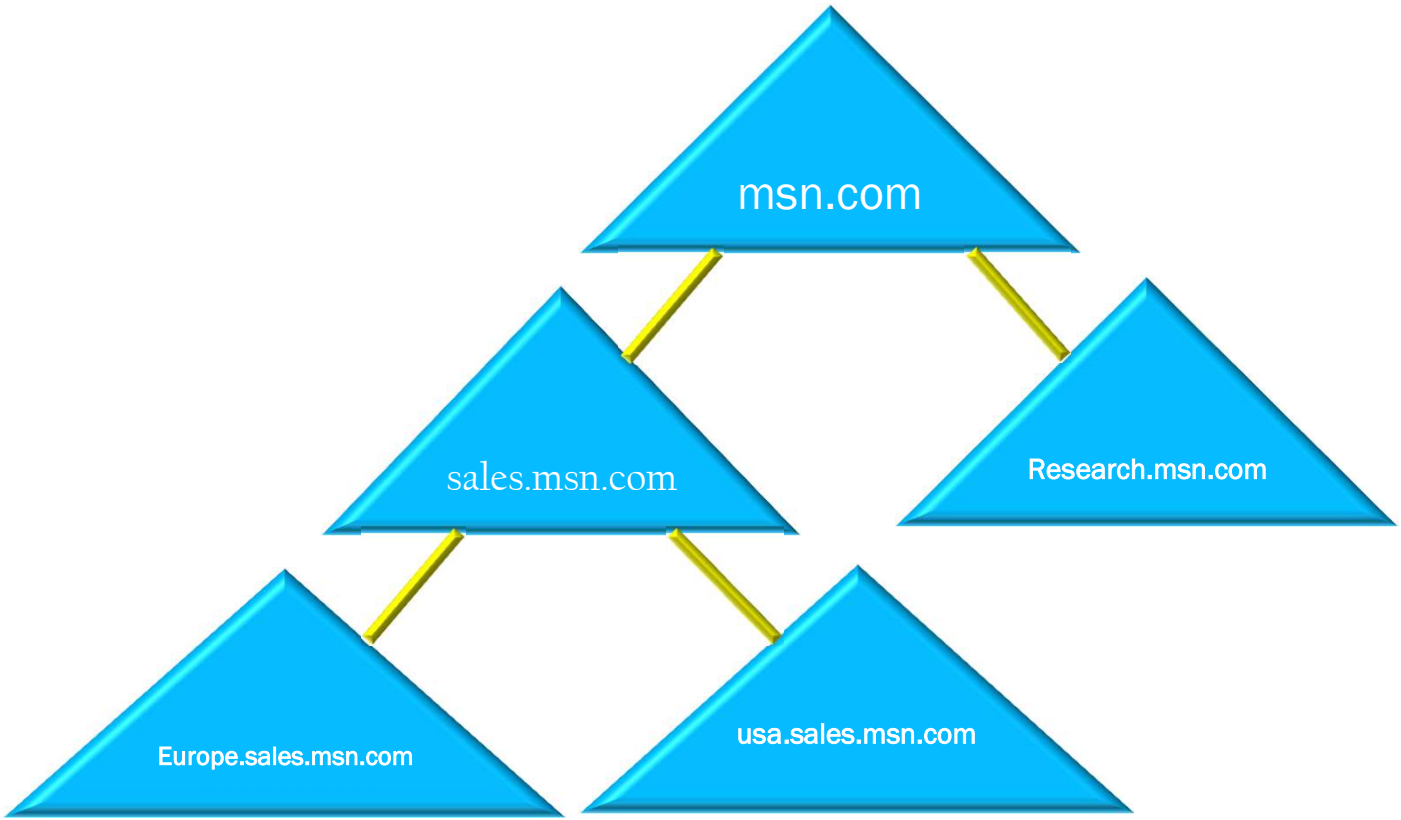
به طور کلی OUها جهت سازماندهی Objectهای درون یک Domain استفاده می شوند.



۴– Tree درخت:

در صورتی که چندین دامین در یک ساختار درختی در کنار هم قرار گیرند اصطلاحاً یک Tree یا درخت را تشکیل خواهند داد.

دامین اولی که ایجاد می کنیم Root domain (دامین ریشه) است و دامین های دیگری که زیر مجموعه آن ایجاد می شوند Child domain می باشند. دامینی که Child domain به آن متصل می شود را اصطلاحاً دامین والد یا Parent نیز می گویند.

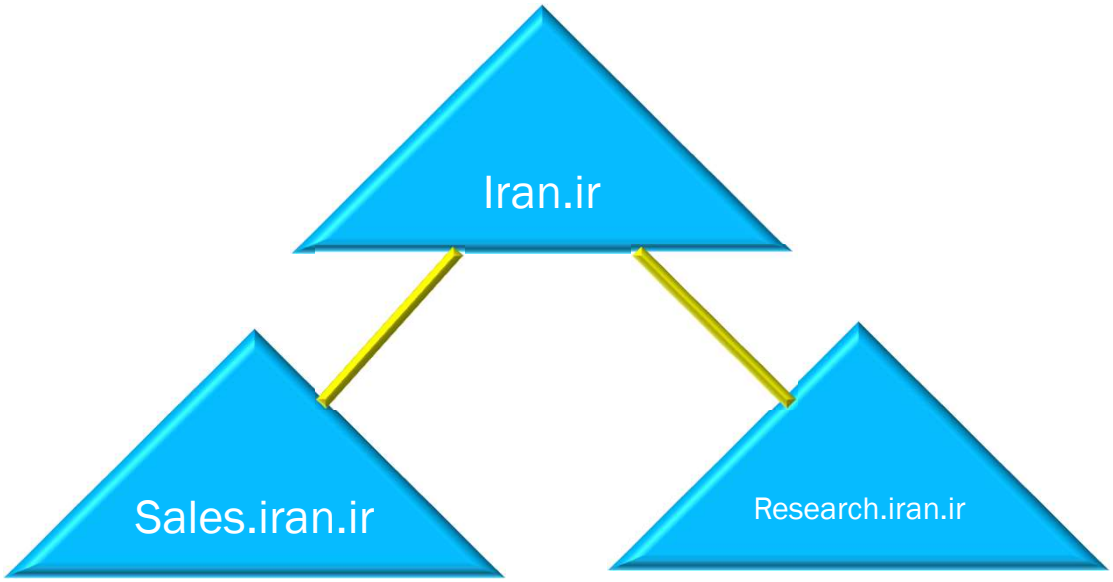


تقسیم بندی های فوق براساس ساختار تجاری شما ایجاد خواهد شد، یعنی در هنگام طراحی شما نگاه می کنید که آیا تجارت شما ایجاب می کند که این بخش ها را به شکل Domainاز یکدیگر جدا نمایید و یا اینکه بخش های مختلف به شکل OU های مختلف داخل یک Domain از یکدیگر جدا شوند.

نکته: تمامی دامین های داخل یک Tree ، Schema ی مشترک دارند که این Schemaی مشترک در Root domain ذخیره می شود.

نکته: به بیانی دیگر درخت به مجموعه ای از دامنه ها با فضای نام مشترک گفته می شود (بخش ریشه در فضای نام همه دامنه ها یکسان است).

برای تشکیل نام کامل دامین، نام Child domain با نام دامین والد ترکیب می شود. برای مثال یک شرکت یک دامین می سازد و نام آن را iran.ir می گذارد. مدیر سیستم بعد از آن تصمیم می گیرد که دو دامین به نام های Sales و Research به آن اضافه نماید.



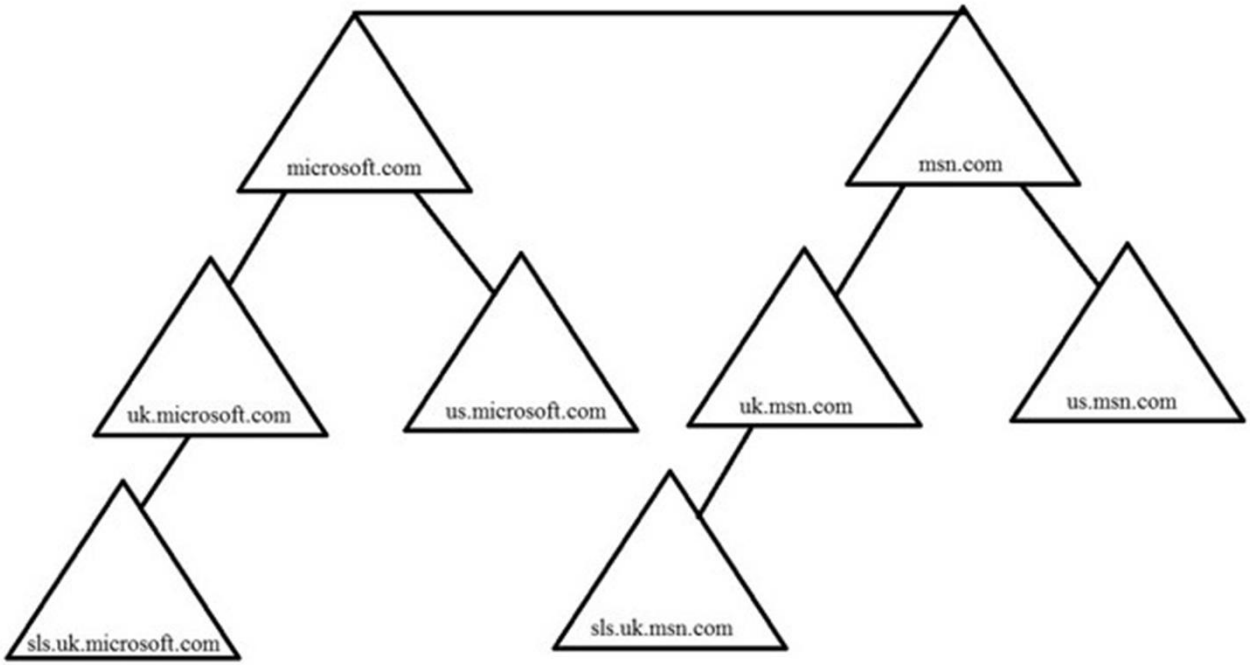
۵– جنگل Forest :

هنگامی که مجموعه ای از Tree ها در یک ساختار درختی در کنار هم قرار می گیرندتشکیل یک forest یا جنگل را می دهند.

اولین Domain که در Forest ساخته میشود

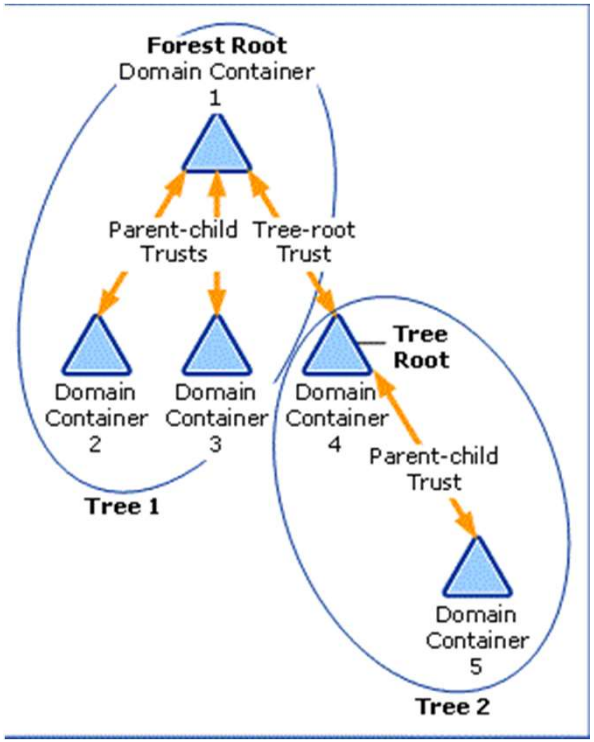
Forest Root Domain نام دارد و نام آن بعنوان نام Forest محسوب میشود.

تمامی Tree های داخل یک Forest ، Schema مشترک دارند که در Root Domain اولین Tree ایجاد شده در Forest ذخیره می شود.



Trust strategy

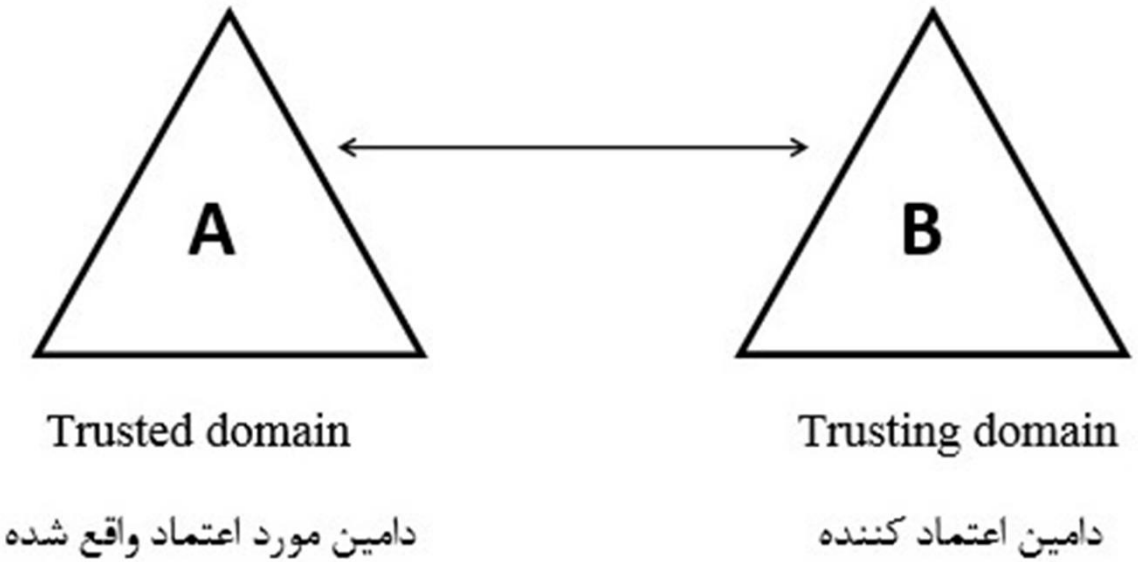
یکی از مسائل مهمی که در هر سازمان یا شرکتی باید در نظر گرفته شود نحوه محافظت از منابع اطلاعاتی است و یکی از مواردی که هر مدیر شبکه ای با آن مواجه میشود تلاش بر این است که منابع آن سازمان یا شرکت برای افراد غیر مجاز قابل دسترس نباشد. از طرف دیگر هر مدیر شبکه ای تلاش می کند که افراد مجاز به راحتی بتوانند به منابع شبکه آن سازمان یا شرکت دسترسی پیدا نمایند.



امکانی که وظیفه فوق را به خوبی انجام میدهد Active directory domain and trustاست. با استفاده از این امکان مدیران شبکه می توانند بین دامین های مختلف (دامین هایی که حوزه های مدیریتی و امنیتی متفاوتی دارند) ارتباط برقرار نمایند که این ارتباط به کاربران یک Domain (مثلاً دامین A) اجازه استفاده از منابع دامین دیگر (مثلاً دامین B) را بدون آنکه آن کاربر به عضویت آن دامین(دامین B) در بیاید فراهم می نماید.

در واقع چون کاربر وقتی وارد دامین A شده Username و Password زده و Authenticate شده دیگر نیازی نیست برای استفاده از منابع موجود در دامین B هم Username و Password داشته باشد. با استفاده از این امکان، مدیران شبکه می توانند این اطمینان خاطر را داشته باشند که کاربران در دامین های مختلف بدون اینکه در آن دامین دارای حساب کاربری باشند از منابع آن دامین استفاده نمایند.

Trust در بردارنده مفهوم اعتماد برای اشتراک گذاری اطلاعات است .



اصلی ترین ویژگی های Trust

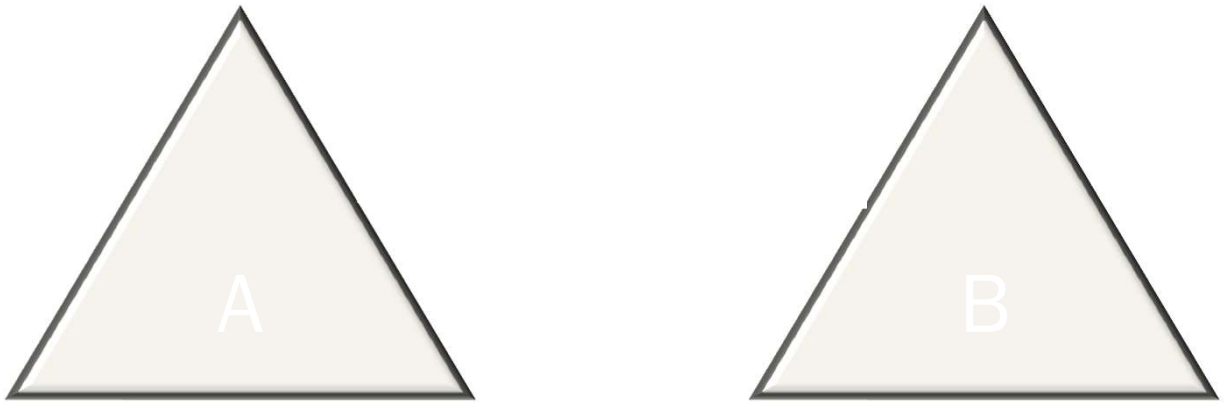
Trustها به طور کلی دو ویژگی اصلی دارند:

۱- جهت اعتماد Trust direction

۲- انتقال پذیری اعتماد Trust transitivity

Trust direction

یک Trust از سمت دامینی که اعتماد می کند به سمت دامینی که به آن اعتماد شده است برقرار می شود. دامین A به دامین B اعتماد دارد. جهت دسترسی همیشه در خلاف جهت اعتماد است. یعنی دامین B به منابع دامین A دسترسی دارد.



(فرض کنید شما به شخص X اعتماد دارید و کلید خانه خود را به آن می دهید پس شخص X می تواند به وسایل خانه شما دسترسی داشته باشد).

Trust ها را هم با جهت یکطرفه و هم با جهت دوطرفه پیاده سازی می نمایند. در Trust یک طرفه شما می توانید در دسترسی منابع یکی از طرفین Trust به دیگری محدودیت هایی را ایجاد نمائید.

انتقال پذیری اعتماد
Trust transitivity

فرض کنید دو tree که parent های آنها با نامهای Domain A و Domain B هستند را داریم و هرکدام از اینها دارای زیرمجموعه ای از دامین ها می باشند، حال می خواهیم بین این دو Tree رابطه Transitive trust برقرار نمائیم. با تعریف Transitive trust بین Domain A و Domain B علاوه بر اینکه این دو دامین به یکدیگر Trustمی کنند، بین کلیه زیر مجموعه Domain A و Domain B نیز رابطه Transitivity trust ایجاد می شود و همینطور بین کلیه دامین های زیر مجموعه Domain A و Domain B . بعبارت دیگر زمانیکه شما بین دو دامین رابطه Transitive trust ایجاد می کنید کلیه دامین هایی که به دامین اعتماد کننده نیز اعتماد دارند به منابع دامینی که به آن اعتماد شده است دسترسی پیدا خواهند کرد.

در واقع Transitive trust شاه کلید دسترسی ها است. زمانیکه Domain A به Domain B اعتماد می کند، به تمامی Child domain های Domain B نیز اعتماد کرده است. حال اگر به هر دلیلی Domain A نخواهد به یکی از Child domain های Domain B اعتماد کند می تواند با اعمال محدودیت هایی یا بلاک کردن دسترسی هایی مانع از دسترسی آن Child domain به منابع Domain A شود.

انواع Trust ها

Default trust : این Trust ها به صورت اتوماتیک ایجاد می شوند

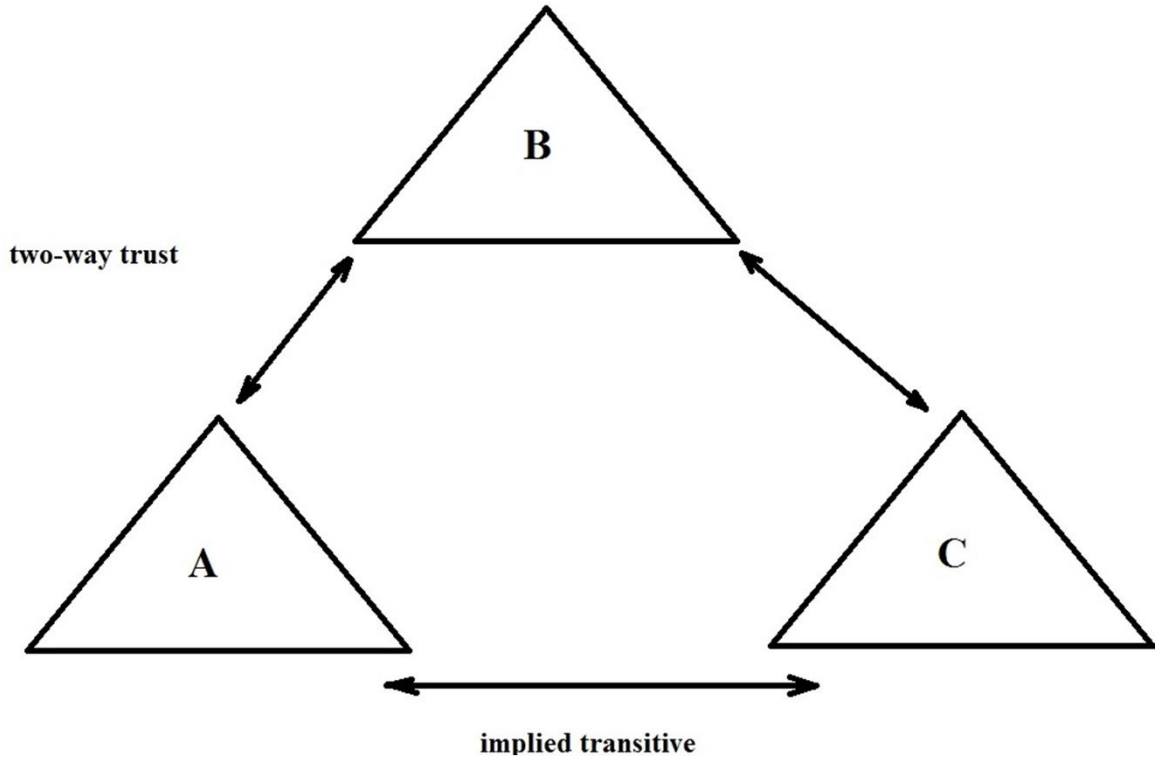
1. Parent and child trust
2. Tree root trust

این Trust ها به صورت دستی ایجاد می شوند

1. External trust
2. Shortcut trust
3. Realm trust
4. Forest trust

این Trust ها می توانند اتوماتیک ایجاد شوند و یا اینکه آنها را دستی ایجاد نمائیم.

Parent and child trust and tree root trust

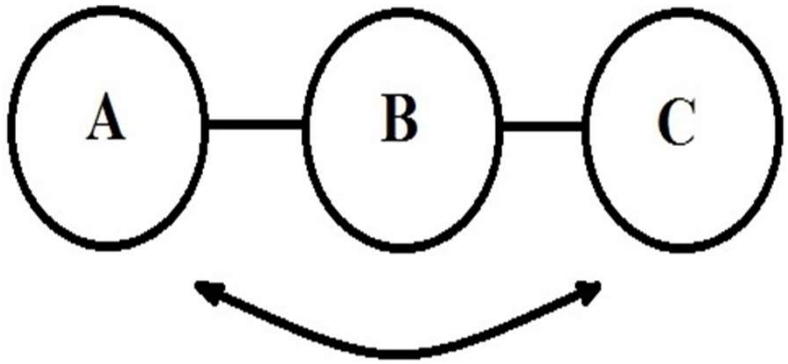


Trust direction

وقتی شما یک Tree ایجاد می نمائید، ابتدا Root domain را ایجاد می کنید بعد زیر مجموعه آن می خواهید دامین های دیگر را ایجاد نمائید. خود به خود این دامین ها به هم Trust می کنند و Trust موجود بین این دامین ها Trust دو طرفه است، یعنی وقتی A به B، Trust کرده B هم به A ، Trust می کند و لزومی ندارد که ما دستی این Trust ها را ایجاد نمائیم. User هایی که در دامین B وجود دارند از امکانات دامین A استفاده می کنند بدون اینکه مجبور باشند یک Username و Password جداگانه ای را ارائه نمایند.

به این حالت Trust دو طرفه Two way trust می گویند و این Trust از نوع Transitive trust است یعنی وقتی دامین اول به دوم Trust داشته باشد. و دامین دوم هم به دامین سوم Trust داشته باشد خود به خود دامین اول به دامین سوم Trust پیدا می نماید که به این حالت Trust transitivity می گویند، یعنی Trust انتقال پیدا می نماید. تمامی دامین هایی که در Forest ایجاد می شوند به این شکل به هم Trust دارند.

نکته: زمانیکه Wizard نصب اکتیو دایرکتوری را برای ایجاد یک دامین جدید در Forestی که از قبل وجود داشته است را اجرا می کنید به طور پیش فرض دو Trust ایجاد شده است. اولین آنها Parent and child trust است و بعدی Tree root trust .

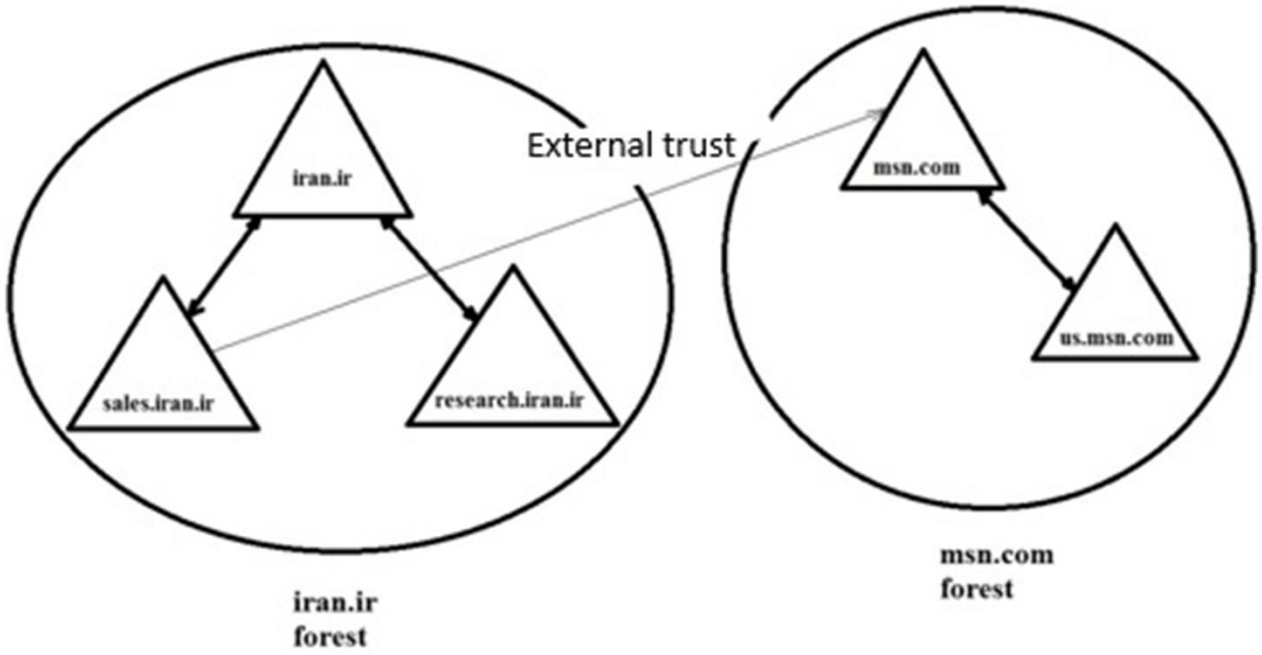


در واقع وقتی یک Parent domain برای خودش یک child domain تعریف می نماید، بین parent domain و child domain به طور اتوماتیک یک Trust ایجاد می شود. در مورد tree root trust هم آشکار است که ریشه های هر درخت به شاخه های خود اطمینان دارند.

نکته: Trustهای پیش فرض یا Default trust به صورت Transitive و two-way تعریف شده اند.

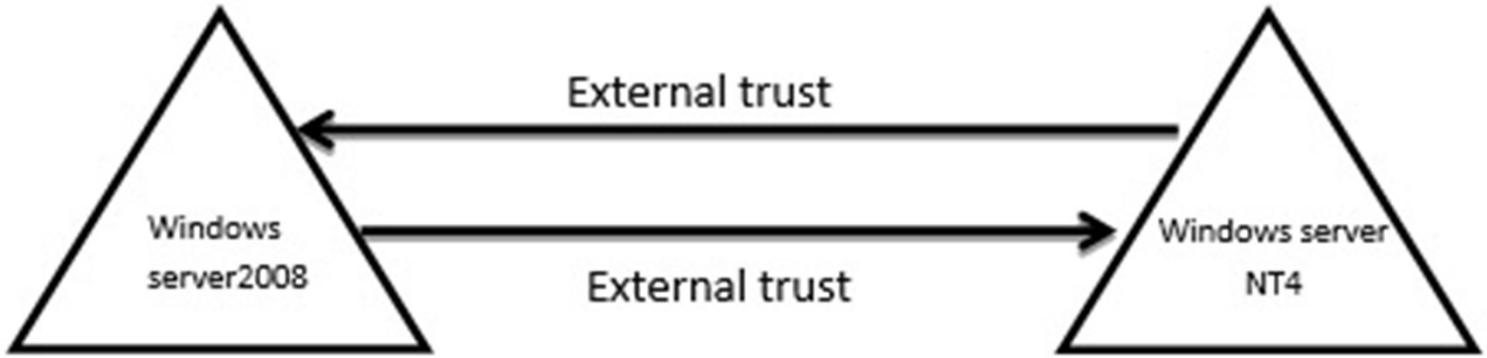
External trust اعتماد های خارجی

برای اینکه بتوانیم با دامین های خارج از Forest ی که در آن قرار داریم رابطه برقرار کنیم از این نوع Trust استفاده می نمائیم.

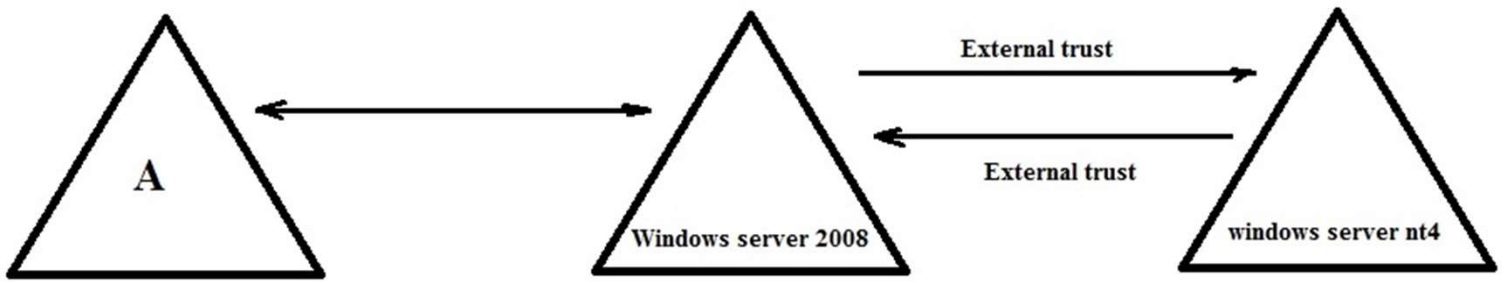


فلسفه ایجاد External trust زمانی است که شما نیازمند دسترسی به منابعی هستید که خارج از سطح forest تان قرار دارند.

به طور مثال اگر دامینی داریم که ویندوز سرور ۲۰۰۸ دارد و می خواهد با دامین دیگر که ویندوز NT4 دارد ارتباط برقرار نماید باید بین آنها External trust ایجاد نمائیم.



- External trust یک Trust یک طرفه است. یعنی برای اینکه یوزرهای NT4 بخواهند از امکانات ویندوز سرور ۲۰۰۸ استفاده نمایند باید یک Trust یک طرفه ایجاد نمائیم و اگر یوزرهای ویندوز سرور ۲۰۰۸ بخواهند از امکانات ویندوز سرور NT4 استفاده نمایند باید یک Trust یک طرفه دیگر هم ایجاد نمائیم. (not bidirectional)
- External trust ، Transitive نیز نیست، یعنی این trust بین دامین های دیگر انتقال نمی یابد.



در شکل بالا دامین A از امکانات ویندوز سرور NT4 نمی تواند استفاده نماید چون External trustی که بین ویندوز سرور ۲۰۰۸ و ویندوز سرور NT4 وجود دارد به دامین A منتقل نمی شود و دامین A باید به طور مستقیم خودش با ویندوز سرور NT4 ، Trust ایجاد نماید. (Not transitive) .

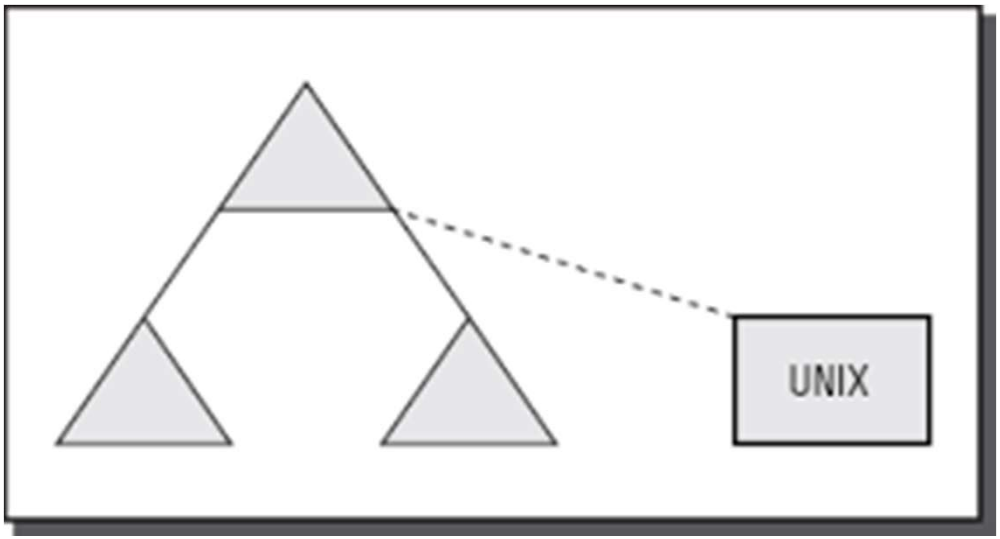
Trust فوق عمدتاً بین دو Domain tree که در یک Forest یکسان قرار دارند ایجاد می شود و فلسفه ایجاد آن ها عمدتاً جهت افزایش سرعت Authentication کردن کاربران و دسترسی به سایر منابع مجاز موجود در بین Domain tree های یک Forest است.

تمامی دامین های موجود در یک forest با هم trust دارند و transitive هم هستند اما وقتی یک یوزری در دامین B می خواهد از امکانات موجود در دامین K استفاده نماید باید چه کاری انجام بدهد؟

باید Identityخودش را به C به D ، E ، F ، G ، H ، I و K برساند و چندین دامین سر راه را بگذراند تا به سر نقطه مقصد برسد. هرکدام از این مراحل یک زمانی میبرد و سرعت را کاهش میدهد و تا شما بتوانید یک منبعی را که نیاز دارید در K بازکنید کلی طول خواهد کشید. برای این کار ما یک Shortcut trust بین B و K ایجاد می نماییم و در نتیجه K مستقیماً با B ارتباط برقرار می نماید و به این ترتیب سرعت افزایش می یابد.

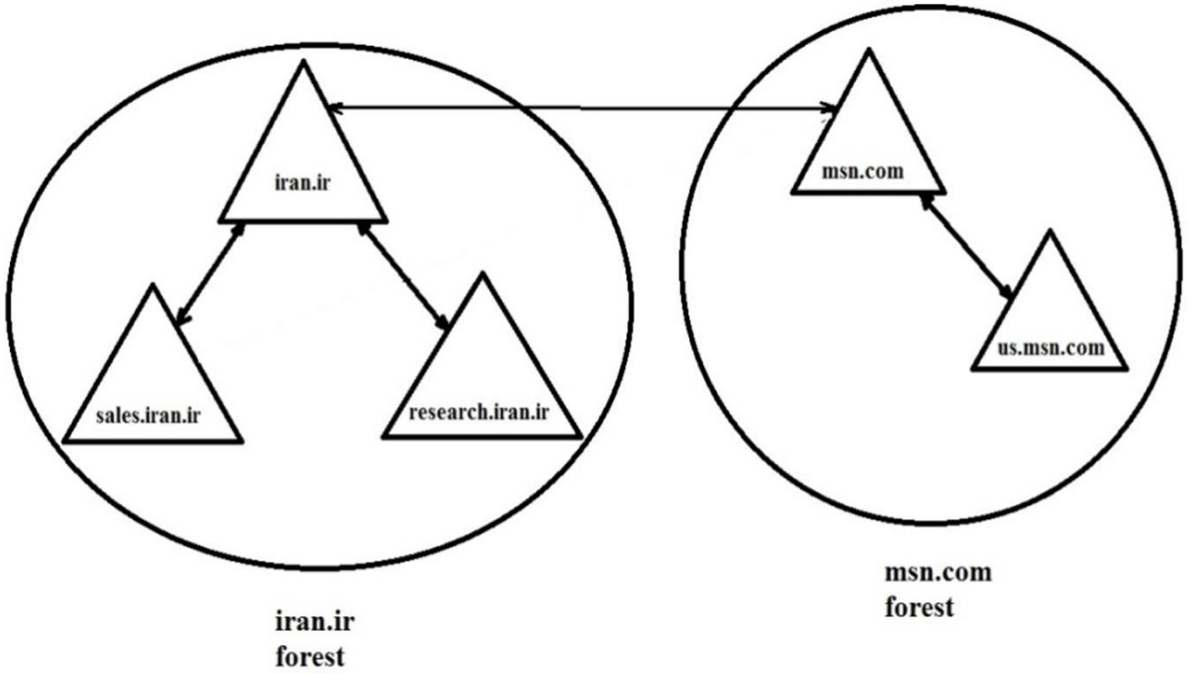
Realm trust

این Trust زمانی کاربرد دارد که می خواهیم بین یک Forestی که سیستم عامل سرور آن غیر مایکروسافتی است (Novel, unix) با یک Forest که سیستم عامل آن مایکروسافتی است Trust ایجاد کنیم.



Forest trust

یک Forest trust یک trust است که بین دو Forest root ایجاد می شود. از این trust زمانی استفاده می شود که لازم باشد بین دو کمپانی با دو جنگل متفاوت همکاری ایجاد شود.



ساختار فیزیکی در اکتیو دایرکتوری

ساختار فیزیکی امکان بهینه سازی و مدیریت ترافیک شبکه را برای شما ایجاد خواهد کرد. این ساختار بصورت فیزیکی وجود دارد و برای انسان قابل لمس است.

ساختار فیزیکی از دو عنصر Domain controller و Site تشکیل شده است.

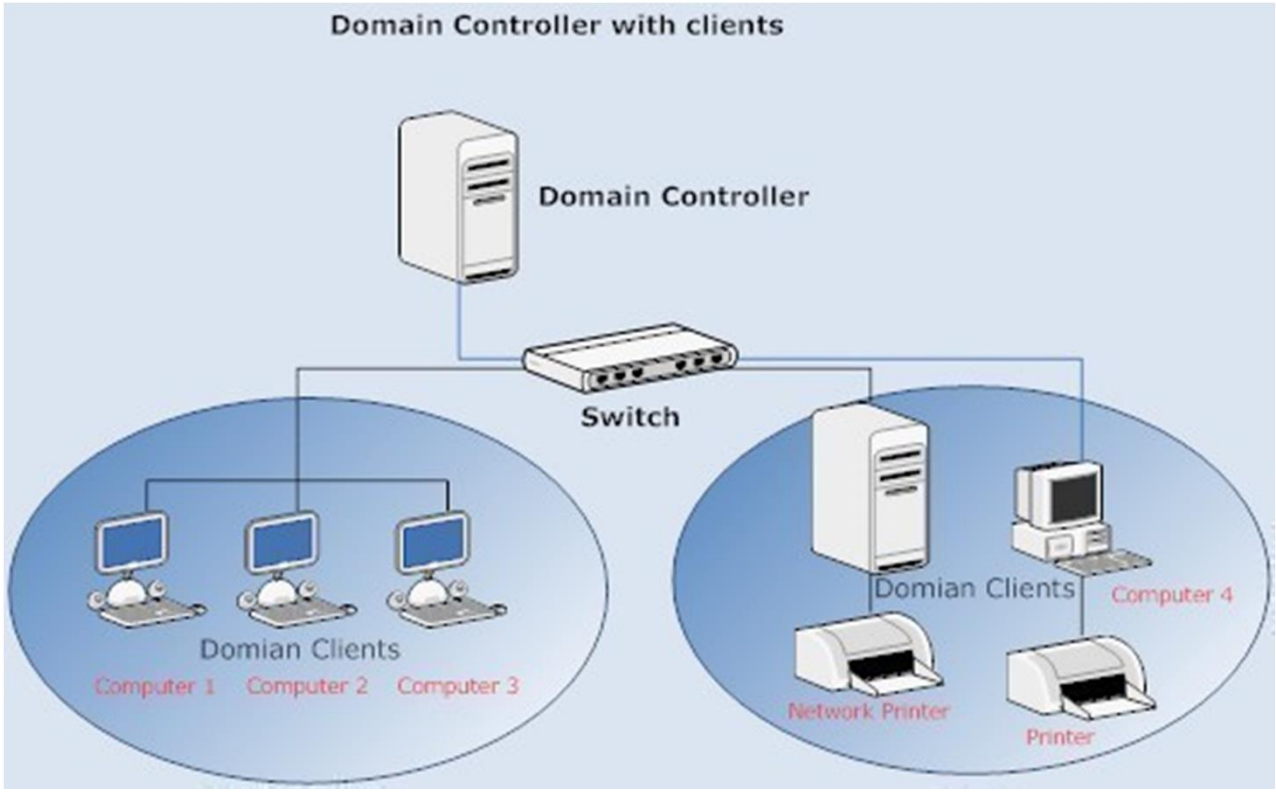
Domain Controller

Domain controller کامپیوتری در شبکه است که بر روی آن سیستم عامل ویندوز سرور نصب شده است و سرویس اکتیو دایرکتوری بر روی آن فعال است.

هر دامین کنترلر فقط می تواند یک دامین را سرویس دهد ولی هر دامین می تواند شامل تعدادی دامین کنترلر باشد.

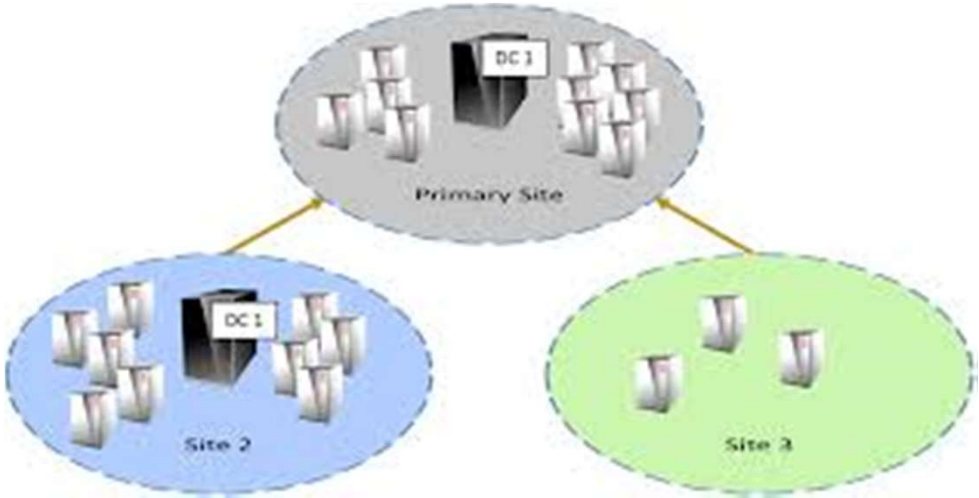
هر دامین کنترلر یک کپی کامل از اطلاعات موجود در اکتیو دایرکتوری رانگهداری می کند. کامپیوتر های دیگری که در داخل دامین قرار دارند پرسش های خود را از دامین کنترلر می پرسند، مثلاً می پرسند که فلان پرینتر کجاست و دامین کنترلر به آنها پاسخ می دهد.

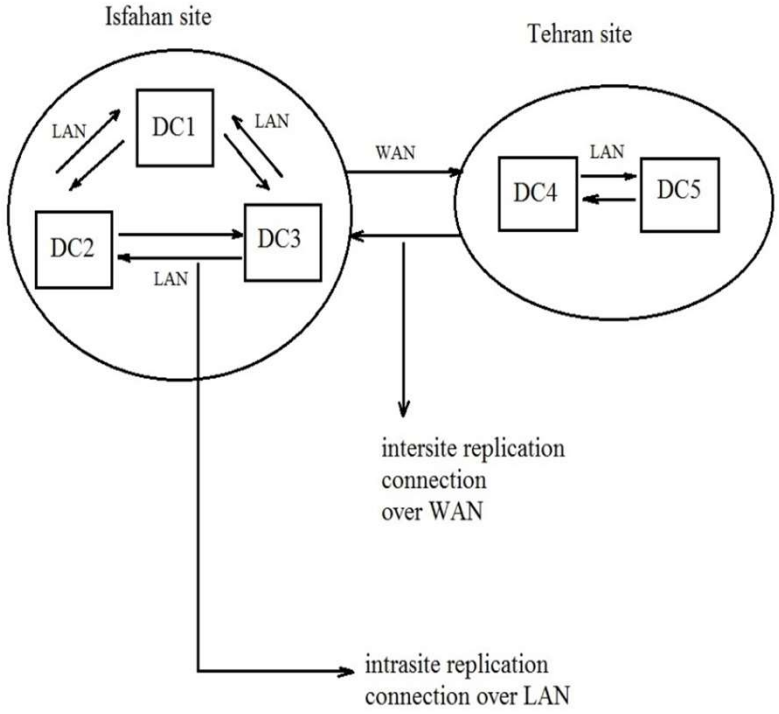
نکته: اگر در شبکه تنها یک DC وجود داشته باشد با از کارافتادن آن، شبکه دامین ما از کار خواهد افتاد و کلاینت ها و کاربران قادر به login در دامین نخواهند بود. برای بر طرف کردن این مشکل می توان از بیش از یک DC در شبکه دامین استفاده نمود تا در صورت از کار افتادن یکی از DC ها همچنان DC دیگر به تقاضاها رسیدگی نماید.



Sites سایت ها

سایت ها مجموعه ای از دامین کنترلرها هستند که در یک یا چند Subnet مختلف قرار گرفته اند و توسط لینک های ارتباطی با سرعت بالا به هم متصل شده اند.(بین سایت ها ارتباطات پرسرعت نیست)





سایت های بالا در دو Subnet مختلف قرار گرفته اند، طبیعتاً این سایت ها دارای محدوده IP متفاوتی هستند.

همانطور که در شکل مشاهده می کنید سایت ها مجموعه ای از کامپیوتر های متصل به هم هستند که با استفاده از لینک های ارتباطی با سرعت بالا به هم متصل شده اند. شما تصور کنید که یک شرکت دارای دفتر مرکزی در تهران و ۲۴ شعبه در شهرستانهای ایران است. این شرکت یک DC دارد که مشابهش باید در هر شهرستان وجود داشته باشد. اینها Domain های متفاوتی نیستند بلکه یک Domain هستند که بین شهرستان ها توزیع شده اند و طبیعی است که هر شهرستان باید یک DC داشته باشد.

حال تصور کنید که هر شهری برای خودش دو تا DC دارد که توسط Link شبکه WAN به شهرهای دیگر و توسط کابل شبکه به DC دیگر در همان شهر متصل شده اند.(دامین کنترلرهای داخل یک سایت توسط کانالهای ارتباطی پرسرعت به هم متصل شده اند ولی بین سایت ها، کانالهای ارتباطی پر سرعت نیستند).

حال تصور کنید که یک Object به اکتیو دایرکتوری یک شهرستان اضافه شود. با توجه به یکپارچه بودن پایگاه داده باید این Object بین تمامی DC های موجود در شبکه تکثیر ، همسان سازی Replicate شود.

1- Intrasite Replication

در داخل سایت ها چون سرعت ارتباطات زیاد است هر تغییری که در یک دامین کنترلر صورت می گیرد خیلی سریع به دامین کنترلر های دیگر منتقل می شود.

2- Intersite Replication

در بین سایت ها چون شبکه WAN است و پهنای باند محدود است و سرعت کانال های ارتباطی پائین است، تغییرات یک DC خیلی سریع به DC دیگر منتقل نمی شود بلکه بر اساس برنامه ریزی این کار انجام می شود مثلاً رأس ساعت ۹ صبح تغییرات یک DC به DC های دیگر اعمال شود.

اینکه Domain Controller از کجا متوجه می شود که آیا اطلاعاتی که قرار است Replicate شود را باید بدون توقف به DC دیگر ارسال نماید یا اینکه باید طبق برنامه زمانبندی شده ارسال نماید دلیل طراحی سایت است. این سایت است که به شما می گوید که الان پهنای باند شبکه ضعیف است و باید صبر کنید و رأس فلان ساعت عملیات Replicate را انجام دهی و یا اینکه می گوید DC در شبکه داخلی شما قرار گرفته است و خیلی سریع می توانید عملیات Replicate را انجام دهید.

نکته: ایجاد سایت در محیط هایی معنا دارد که شبکه گسترده ای دارید که در آن شبکه عملیات زیر شبکه سازی (Subnetting) انجام شده و زیر شبکه ها با لینک WAN بهم متصل شده اند و سرورها باید با هم عملیات Replicate را انجام دهند. در یک شبکه ایجاد سایت چندان معنا ندارد.

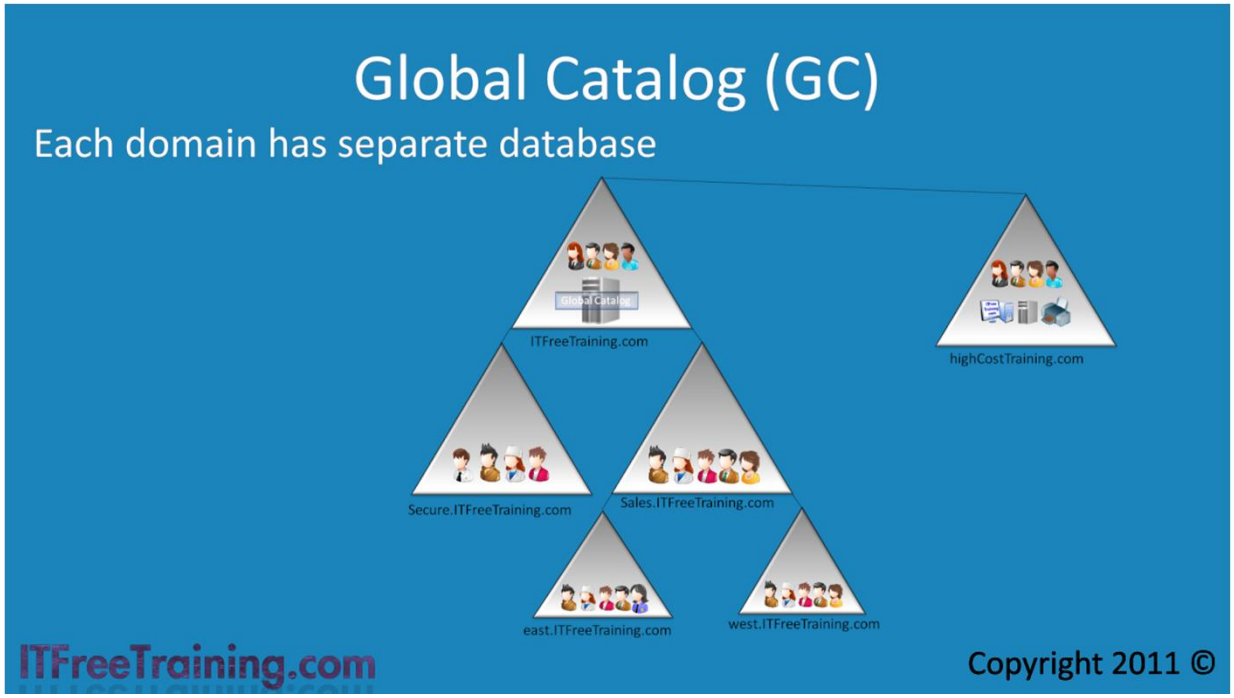
تکثیر و همسان سازی
Replication

تکثیر و همسان سازی انتقال تغییرات بین کنترل کننده های دامین (DC) است زمانیکه شما یک کاربر به شبکه اضافه می کنید یا یک رمز عبوری را تغییر می دهید، به هر حال هر تغییری که در یک DC داده می شود آن تغییر باید به سایر DC ها در دامنه مخابره (تکثیر) شود.

این امر ترافیکی در شبکه ایجاد می نماید. زمانی که شما یک سایت را ایجاد می نمائید، تغییرات بین DC های داخل یک سایت بطوری فوری تکثیر می شود اما تکثیر اطلاعات بین سایت ها قابل زمانبندی و مدیریت است.

Global Catalog (GC)
کاتالوگ عمومی

کاتالوگ عمومی لیستی از تمام اشیاء در جنگل است. این کاتالوگ به منظور جستجوی اشیاء AD استفاده می گردد.



از آنجایی که تنها یک GC می تواند در جنگل وجود داشته باشد و هر جنگل می تواند شامل چندین دامنه باشد بنابراین حجم فایل GC می تواند کمی بزرگ باشد و برای محدود کردن حجم آن، اشیاء داخل GC تنها می توانند دارای زیر مجموعه ای از مشخصات اصلی باشند. بعنوان مثال اگر یک حساب کاربری دارای ۱۰۰ مشخصه برای توصیف آن باشد، تنها ۱۰ مشخصه از آن در GC آورده می شود.

یک جنگل با دو دامنه را تصور کنید. هر دامنه دارای ۲ کنترل کننده دامنه است. اگر یک کاربر در دامنه B یک کاربر را در دامنه A جستجو نماید چه اتفاقی خواهد افتاد ؟

کنترل های دامنه B هیچ یک از اطلاعات شیء های دامنه A را نگه نمی دارند. بنابراین یک کنترل در دامنه B نمی تواند به یک پرس و جو در مورد شیء های دامنه A جواب بدهد. کاتالوگ سراسری قسمتی از اطلاعات هر شیء در جنگل را نگه می دارد. زمانی که یک کاربر در دامنه B دنبال یک شیء در دامنه A باشد GC نتیجه پرس و جو را فراهم می نماید. برای بهینه سازی کارایی GC ویژگی تمام شیء های جنگل را نگه نمی دارد در عوض شامل یک زیر مجموعه از ویژگی هاست که برای جستجو در دامنه مفید هستند.

مبانی سرویس دهنده نام های حوزه
Domain Name System (DNS)

سرویس Domain Name System یا سیستم نام دامنه، که به اختصار DNS نامیده می شود.

هر کامپیوتر بر روی اینترنت دارای هم نام (Host Name) و هم آدرس IP می باشد و برای دسترسی به یک کامپیوتر در شبکه هم می توان از نام کامپیوتر استفاده نمود و هم امکان استفاده از IP آن کامپیوتر وجود دارد. ولی اکثراً ترجیح می دهند که به جای استفاده از اعداد و ارقام آدرس IP ، از نام رایانه استفاده نمایند چرا که به خاطر سپردن نام به مراتب راحت تر از آدرس IP می باشد.

به عنوان مثال با استفاده از آدرس Ipv4 مربوط به سازمان سنجش آموزش کشور که برابر ۹۲.۲۴۲.۱۹۵.۱ است و وارد کردن این عدد IP در مرورگر وب، می توان وب سایت سازمان سنجش را مشاهده نمود ولی با استفاده از www.sanjesh.org هم می توان به آن دسترسی پیدا کرد.

ترجمه نام به آدرس IP را اصطلاحاً Name Resolution می گویند.

Hosts file

در سالهای نخستین راه اندازی شبکه ARPANET ، راه حلی بسیار ساده برای ترجمه نام کامپیوتر به آدرس IP وجود داشت و آن تعریف تمام نامها و آدرس های IP معادل در یک فایل به نام hosts.txt بود. در آن روزگار تعداد ماشین های میزبان زیاد نبود و حجم چنین فایلی چندان بزرگ نمی شد. هر ماشین میزبان رأس ساعت ۱۲ هر شب این فایل را تازه سازی و به روز می کرد تا هرگونه تغییر احتمالی و تعریف آدرس های جدید اعمال شود.

این روش مشکلاتی داشت از جمله اینکه :

الف) با ازدیاد کامپیوتر ها تعداد آدرس ها و نام ها زیاد شده و بعضاً نام ها با هم در تداخل افتاده و تضخیص اینکه یک نام وجود دارد یا خیر کار مشکلی است.

ب) فایل host.txt فقط بر روی یک کامپیوتر موجود بود و اگر کاپیوتر های دیگر می خواستند که آن فایل را دانلود نمایند باید مدام تغییرات لحظه به لحظه آن فایل را دنبال می کردند و این کار مشکلی بود.

ج) تمامی بار عملیات مربوط به Name Resolution بر روی یک کامپیوتر است و اگر برای لحظه ای کوتاه آن کامپیوتر قابل دسترس نباشد عملیات Name Resolution خاتمه می یابد.

بدیهی است که داشتن یک فایل متمرکز و قراردادن تمام آدرس ها و معادل آدرس IP در آن، امروزه با حجم چند صد میلیونی آدرس ها در اینترنت امکان پذیر نیست و سالهاست از روشی برای تبدیل نام کامپیوتر به آدرس IP استفاده می شود که DNS نام دارد.

DNS رش سلسله مراتبی است که بانک اطلاعاتی مربوط به نام کامپیوترها و معادل IP آنها را روی کل شبکه اینترنت توزیع کرده است. هر کامپیوتر در شبکه بخشی از عملیات Name Resolution را انجام می دهد. در واقع هر کامپیوتر در شبکه می تواند در یک روال منظم و سلسله مراتبی آدرس IP معادل با کامپیوتر مورد نظرش را در هر نقطه از شبکه پیدا نماید.

در DNS کل آدرس های اینترنت درون بانکهای اطلاعاتی توزیع شده ای هستند که هیچ تمرکز روی نقطه خاصی از شبکه ندارند. روش ترجمه نام بدین صورت است که وقتی یک برنامه کاربردی مجبور است برای برقراری یک ارتباط معادل آدرس IP از یک کامپیوتری به نام sanjesh.org را بدست بیاورد قبل از هر کاری یک تابع کتابخانه ای را صدا می زند. به این تابع کتابخانه، تابع تحلیل گر نام می گویند. تابع تحلیل گر نام آدرس نمادینی را که بایستی ترجمه شود به عنوان پارامتر ورودی پذیرفته و سپس یک بسته درخواستی به روش UDP تولید کرده و به آدرس یک سرویس دهنده DNS (که به صورت پیش فرض مشخص است) ارسال می نماید.

همه ماشین های میزبان حداقل باید آدرس IP از یک سرویس دهنده DNS را در اختیار داشته باشند. این سرویس دهنده محلی پس از جستجو، آدرس IP معادل با یک نام کامپیوتر را بر می گرداند. تابع تحلیل گر نام نیز آن آدرس IP را به برنامه های کاربردی تحویل می دهد. با پیدا شدن آدرس IP ، برنامه کاربردی می تواند عملیات مورد نظرش را ادامه دهد.

بررسی روش های جستجو در سیستم DNS

همانگونه که اشاره شد، بانک اطلاعاتی که نام کامپیوتر ها را با معادل IP آنها در خود دارد متمرکز نیست، بلکه روی کل اینترنت توزیع شده است. حال باید دید نام کامپیوترها چگونه سازماندهی می شود تا نهایتاً بتوان روش جستجو روی یک بانک اطلاعاتی توزیع شده را توضیح داد:

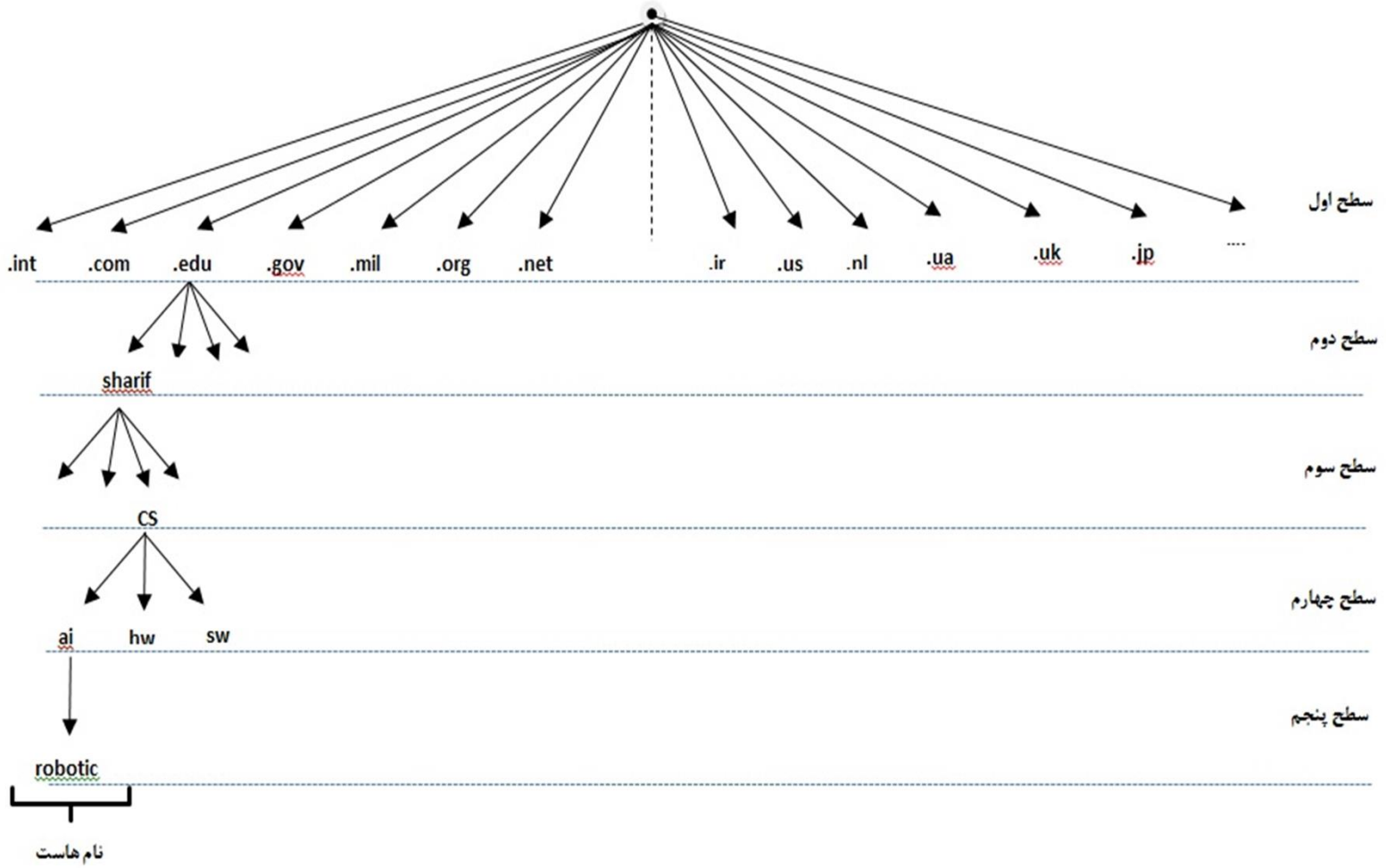
- نام میزبان در شبکه های کامپیوتری به نام نمادین FQDN مشهور است.
- Full Qualified Domain Name (FQDN)

Full Qualified Domain Name (FQDN)

اسامی نمادین زیر را در نظر بگیرید:

- www.president.ir
- www.sharif.edu
- www.microsoft.com
- www.refahi.gov.ir
- www.robotic.ai.cs.sharif.edu

بدیهی است که نامهای نمادین مانند مثالهای بالا بدون دلیل انتخاب نشده اند بلکه اطلاعاتی ارزشمند برای جستجو در بانک اطلاعاتی توزیع شده ی نام های نمادین در خود دارند. به گونه ای که مشهود است یک نام حوزه از چندین بخش مجزا که با علامت "." از هم تفکیک شده تشکیل می شود. هر کدام از این بخش ها که سطح (level) نام دارد به یک قسمت از بانک اطلاعاتی توزیع شده اشاره می کند و به محدود کردن فضای جستجو می انجامد.


سطح اول
.int
.com
.edu
.gov
.mil
.org
.net
.ir
.us
.nl
.ua
.uk
.jp
....
سطح دوم
sharif
سطح سوم
CS
سطح چهارم
ai
hw
sw
سطح پنجم
robotic
نام هاست

در نمودار اسلاید قبل چگونگی شکسته شدن یک آدرس به زیر حوزه های کوچکتر به تصویر کشیده شده است. با این ساختار برای ترجمه ی یک نام حوزه مثل robotic.ai.cs.sharif.edu ، عملیات از محلی به نام ((ریشه)) شروع می شود. پس آدرس ماشین که راهنمای آدرس های edu. در آن جا واقع شده است بدست می آید. با مراجعه به چنین ماشینی مجدداً آدرس ماشین دیگر که راهنمای آدرسهای sharif.edu. در آنجا قرار دارد بدست می آید. این روند تا رسیدن به ماشینی که آدرس IP معادل را در اختیار داشته باشد ادامه می یابد. این عملیات در چند مرحله ی محدود تکرار می شود و با توجه به آنکه در این پرس و جو از پروتکل UDP استفاده شده تاخیر چندانی نخواهد داشت.

Domain name space

اسامی در DNS از ساختار درختی سلسله مراتبی به اسم فضای نام دامنه یا Domain name space تشکیل شده اند و از پنج مجموعه برای تشریح فضای نام دامنه استفاده می شود:

الف) Root Domain

در بالاترین محل ساختار درختی دامنه، ریشه Root Domain قرار دارد و به صورت یک نقطه "."می باشد، یعنی تمامی اسامی اینترنتی به یک نقطه ختم می شوند.

ب) Top Level Domain (TLD) :

دومین بخش از ساختار درختی یک آدرس می باشد که تعیین کننده حوزه فعالیت و حوزه جغرافیایی می باشد و به دو بخش تجزیه می شود:

ب-۱) Generic TLD (GTLD) :

به معنی حوزه عمومی فعالیت و تعیین کننده نوع سازمان بوده و شامل پسوندهای سه حرفی زیر است:

com.صاحب این نام جزو موسسات اقتصادی و تجاری است.

edu.صاحب این نام جزو موسسات علمی یا دانشگاهی به شمار می آید.

gov.این مجموعه از نام ها برای آژانس های وابسته به دولت به کار می رود.

int.صاحب این نام یکی از سازمانم های بین المللی (مثل یونسکو، یونیسف، فائو و ....) است.

mil.صاخب این نام یکی از سازمان های نظامی دنیا به شمار می رود.

net.صاحب این نام جزو یکی از ارائه دهندگان خدمات شبکه است.

org.صاحب این نام جزو یکی از سازمان های عام المنفعه و غیر انتفاعی می باشد.

ب-۲) Country Code TLD (CCTDL) :

در این قسمت از یک استاندارد دو حرفی برای تعیین کشور (حوزه جغرافیایی) استفاده می شود مانند:

Ir ایران

Ca کانادا

Nl هلند

Jp ژاپن

Us ایالات متحده امریکا

Ua امارات متحده عربی

Tw تایوان

Iq عراق

به عنوان مثال آدرس www.bmi.ir آدرس اینترنتی بانک ملی جمهوری اسلامی ایران است.

در بعضی مواقع GTLD و CCTLD به صورت ترکیبی مورد استفاده قرار می گیرند به طوریکه ابتدا GTLD و سپس CCTLD قرار می گیرد.

ج) Secondary Level Domain (SLD) :

دومین سطح دامنه می باشد که به صورت منحصر به فرد می باشد. در واقع شما نمی توانید هر نام دلخواهی را برای شرکت یا سازمان خودتان انتخاب نمائید بلکه برای این کار باید نام مورد نظر را ثبت نمائید، در غیر اینصورت چنین آدرسی در اینترنت هویت نخواهد داشت. برای ثبت آدرس باید به سایت های ثبت نام کننده نام حوزه مراجعه کرده و تقاضای ثبت آدرس نمائید. این موسسات ضمن ثبت و درج نام در بانک اطلاعاتی یکی از حوزه های سطح بالا (در واقع نام مورد نظر زیر مجموعه یکی از TLD ها قرار می گیرد) تضمین خواهند نمود که نام انتخابی شما در کل شبکه ی اینترنت منحصر بفرد و قابل شناسایی است.

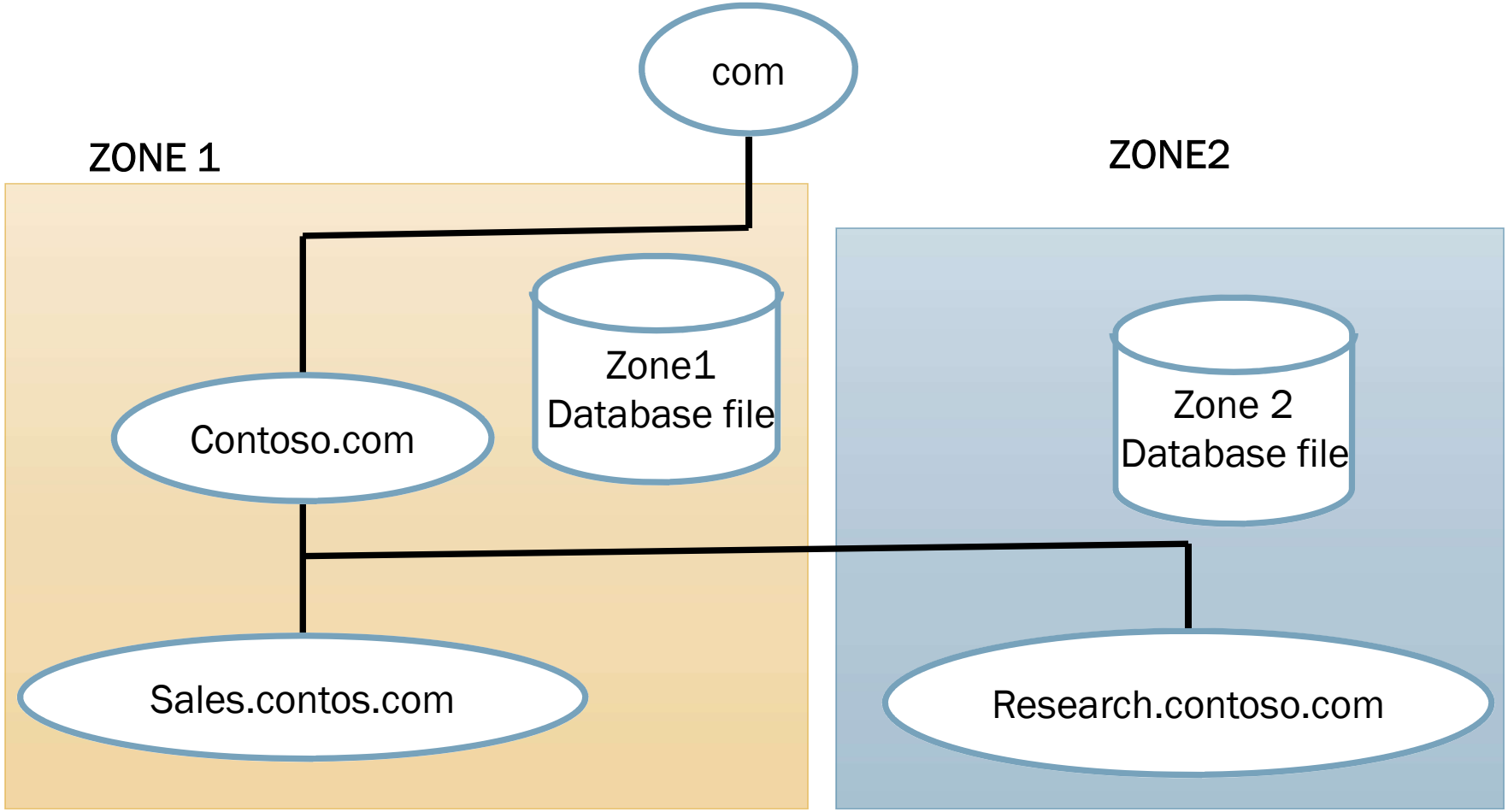
نکته:

Mailserver.sales.cotoso.com اسم کامل یک کامپیوتر شامل نام کامپیوتر، نام sub domain ها و نام domain و نام top level domain می باشد و در واقع این نام کامل نشان می دهد که هر host در کجای این درخت قرار دارد و ما می توانیم از این طریق آن را پیدا کنیم که به این نام کامل FQDN (Full Qualified Domain Name) می گویند.

اجزای DNS

۱- Zone

بخشی از فضای نام دامنه (Domain Name Space) در DNS است. یک Zone فضای نام دامنه را به چند بخش تقسیم می کند تا مدیریت آنها برای مدیران راحت تر شود. باید توجه داشت که Zone معادل domain نیست بلکه از یک یا چند domain مجاور (domain هایی که زیر مجموعه هم باشند) تشکیل شده است. (چند domain غیر مجاور نمی توانند یک zone را تشکیل دهند.


com
ZONE 1
ZONE2
Zone1
Database file
Contoso.com
Zone 2
Database file
Sales.contos.com
Research.contoso.com

نکته:در شکل اسلاید قبل یک zone می تواند دامین com. را در بر گیرد و یا یک zone می تواند com. و زیر مجموعه های com. مانند research.contoso.com و sales.contoso.com را هم در بر گیرد.

اما research.contoso.com و sales.contoso.com نمی توانند در یک zone قرار گیرند چون زیر مجموعه هم نیستند. در واقع ما می توانیم یک domain تک را در داخل یک zone قرار دهیم و یا یک domain را با subdomain داخل یک zone قرار دهیم اما نمی توانیم دو تا subdomain را داخل یک zone قرار دهیم.

نکته: هر zone شامل یک بانک اطلاعاتی مخصوص به خود است که تمامی اطلاعات مربوط به زیر دامنه های خود را در آن نگهداری می کند.

۲- Name server

به کامپیوتری گفته می شود که سرویس DNS بر روی آن نصب شده باشد و داده های مربوط به یک یا چندین zone را در خود نگهداری نماید. در واقع name server دارای یک بانک اطلاعاتی اصلی است که به بانک اطلاعاتی zone اشاره دارد.

نکته: هر zone حداقل یک name server دارد.(یک کامپیوتر وقتی به دنبال کامپیوتر دیگری در شبکه می گردد از name server آدرس آن کامپیوتر را می پرسد) و البته وجود چند name server در شبکه برای مواقعی که یک name server خراب باشد و قادر به پاسخگویی نباشد مناسب است و در این مواقع می توان به name server های دیگر مراجعه نمود.

انواع zone

الف- Active Directory Integrated

فقط در شرایطی که شما روی یک کامپیوتر active directory را نصب کرده باشید و DNS هم روی آن کامپیوتر موجود باشد این قضیه اتفاق می افتد یعنی اطلاعات DNS داخل active directory ذخیره می شود

(replication ، DNS هم با Replication اکتیو دایرکتوری به صورت همزمان انجام می شود بنابراین بار کاری خیلی کم می شود)

ب- Primary zone :

وقتی برای اولین بار در DNS یک zone می سازید باید از نوع primary باشد. در واقع در هر ساختار DNS حداقل یک zone به صورت primary وجود دارد. این zone هم خواندنی و هم نوشتنی است و این بدان معناست که مدیر سیستم قادر به اضافه کردن، حذف و ویرایش رکوردها و تغییر تنظیمات مربوط به این نوع از zone می باشد.

ج- Secondary zone

Secondary zone ها یک نسخه فقط خواندنی از primary zone می باشند بدین معنی که پس از ایجاد primary zone می توان یک کپی از اطلاعات آن را که فقط امکان خواندن از آن وجود دارد در اختیار سایر سرورهای DNS قرار داد تا آنها بتوانند به درخواست هایی که از طرف کاربران داده می شود پاسخ دهند. استفاده از این zone ها بیشتر در مواردی است که بار زیادی بر روی سرور DNS اصلی شبکه قرار داشته باشد. در این مورد می توان سرورهای DNS کمکی اضافه نمود و نسخه ای از primary zone را (با استفاده از secondary zone) در اختیار آنها قرار داد. با این کار بار شبکه بر روی این سرورها توزیع می گردد. قبل از ایجاد secondary zone بر روی یک سرور DNS این سرور باید برای سرور اصلی (که primary zone را در اختیار دارد) شناخته شود.

د- Stub zone

Stub zone در واقع یک کپی از یک primary zone است که صرفاً حاوی رکورد های ضروری برای شناسایی آن zone می باشد. Stub zone تا حدود زیادی با secondary zone شباهت دارد با این تفاوت که در secondary zone کلیه رکوردهایی که در primary zone قرار دارد وجود دارد اما در stub zone صرفاً حاوی رکوردهای ضروری است برای شناسایی سرورهایی که رکوردهای مورد نظر را درون خود نگه می دارند.

اجزای DNS

Resource record

بانک اطلاعاتی zone ، اطلاعات خود را به صورت رکورد ذخیره می نماید. این رکوردها به صورت متفاوتی اطلاعات را نگهداری می نماید که مهمترین آنها عبارتند از:

الف- Aیک نام FQDN را می گیرد و یک آدرس IPV_4 را بر می گرداند.

ب- AAAAیک نام FQDN را می گیرد و یک آدرس IPV_6 را بر می گرداند.

SOA (Start Of Authority) :

SOA اولین رکوردی است که به صورت خودکار در یک zone ساخته می شود و یک سری اطلاعات ابتدایی در خصوص zone در آن قرار گرفته است و قابل حذف نیست. اطلاعاتی مانند نحوه ی انتقال اطلاعات بین zone اصلی و zone های ثانویه، (به طور مثال بازه زمانی به روزرسانی به صورت پیش فرض ۱۵ دقیقه یک بار است و در صورت عدم موفقیت هر ۱۰ دقیقه یک بار تلاش مجدد صورت می گیرد. بدون پاسخگویی یک سرور اصلی به صورت پیش فرض یک سرور ثانویه تا یک روز اطلاعات نسبتاً نعتبری دارد و به query ها پتسخ می دهد) یک شماره سریال، مدیر مسئول (مسئول تعریف اسامی) و ...

ادامه SOA (Start Of Authority) :

Start of authority RR

Tpci.com IN SOA merlin.tpci.com

serial number 2) root.merlin.tpci.com

7200	refresh (2hrs)
3600	retry (1hr)
151200	expire (1 week)
86400	min TTL

Cname

نامهای مستعار را برای یک آدرس تعیین می نماید. بعنوان مثال شاید کسی حدس بزند آدرس شبکه ی دانشکده کامپیوتر در دانشگاه MIT به صورت cs.mit.edu است اما نمیداند آدرس اصلی به دلایلی به صورت lcs.mit.edu تعریف شده است که کمتر قابل حدس و بخاطر سپردن است. برای رفع این مشکل کوچک می توان در فایل RR نام مستعار cs.mit.edu را معادل با نام اصلی تعریف کرد.

Cs.mit.edu 86400 IN CNAME lcs.mit.edu

از طرفی اسامی نمادین گاهی طولانی هستند (مثل www.altavista.com) و صاحبان این نامها ترجیح می دهندبرای از دست ندادن مراجعین کم حوصله اسامی کوتاه با نام اصلی داشته باشند (مثل www.av.com) در چنین حالتی این رکورد بکار می رود:

www.av.com 86400 IN CNAME www.altavista.com

نکته: وقتی DHCP یک آدرس را به کامپیوتری می دهد آن کامپیوتر به DNS درخواست می دهد که من یک آدرس دارم و یک نام هم دارم و این نام و آدرس را برای من ثبت کن (مثل ثبت اطلاعات اولیه در ۱۱۸) پس DNS اطلاعات را در resource record ذخیره می نماید

روش های جستجو در سرویس دهنده های نام:

همانطور که قبلاً اشاره شده است اسامی نمادین در شبکه ی اینترنت در قالب حوزه ها و زیرحوزه ها سازماندهی شده اند و در یک فایل یکتا متمرکز نیستند بلکه روی کل شبکه اینترنت توزیع و مدیریت آنها به صورت غیر متمرکز تفویض شده است. به همین دلیل برای ترجمه یک نام به آدرس IP ممکن است چندین مرحله پرس و جو صورت گیرد تا یک آدرس پیدا شود.

سه روش برای پرس و جوی نام در سرویس دهنده های نام وجود دارد:

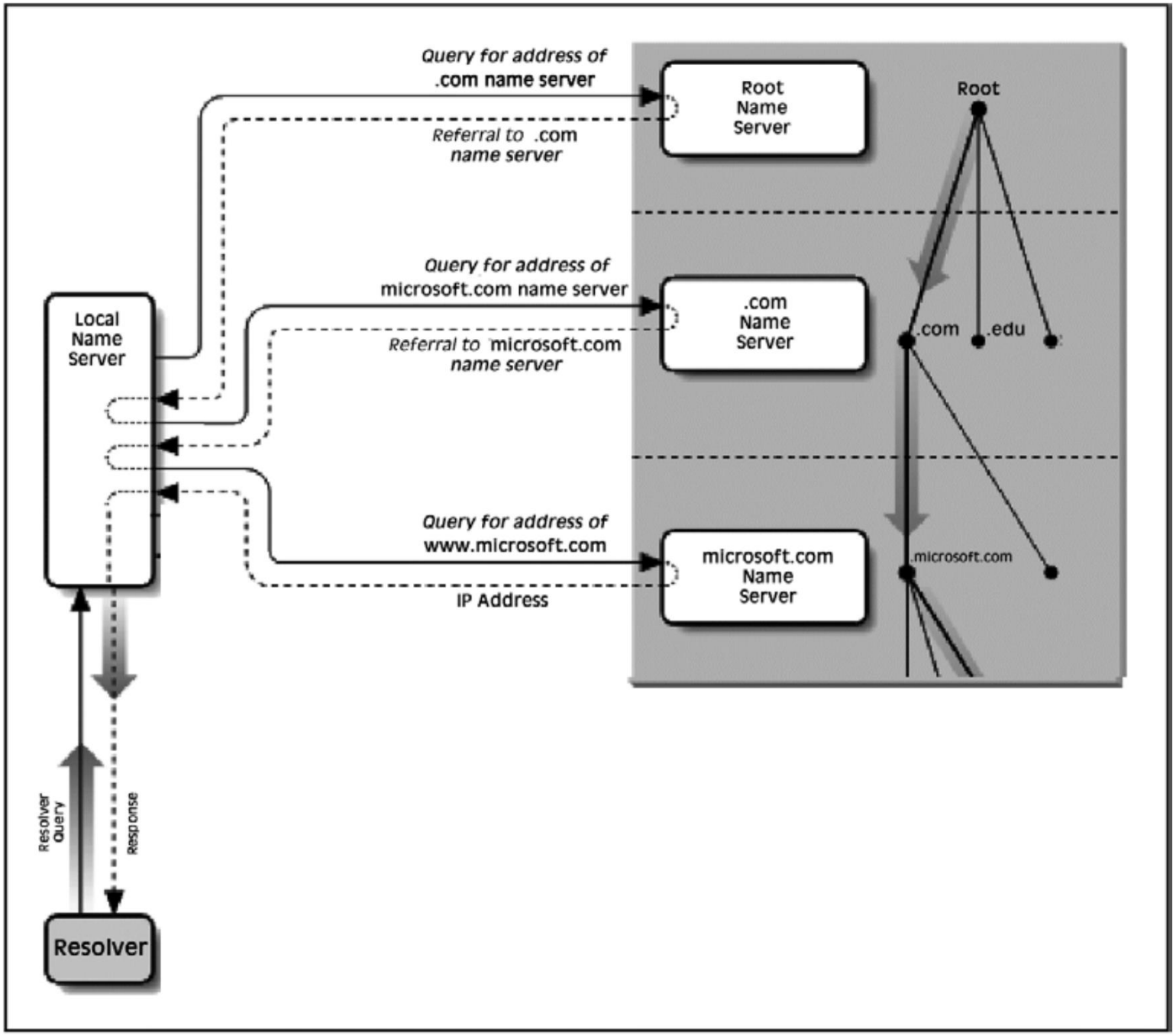
1. پرس و جوی تکراری
2. پرس و جوی بازگشتی
3. پرس و جوی معکوس

پرس و جوی تکراری

در پرسوجوهای تکراری قسمت اعظم تلاش برای تبدیل یک نام برعهدهٔ سرویس دهندهٔ محلی است؛ اینDNS حداقل به آدرس ماشینRoot به عنوان نقطهٔ شروع نیاز دارد وقتی یک تقاضای ترجمه آدرس به سرویسدهندهٔ محلی ارسال میشود در صورتی که قادر به ترجمهٔ نام به معادل IPآن باشد معادل آدرس IPنام مورد نظر را به تقاضا کننده برمیگرداند. این حالت وقتی است که سرویس دهندهٔ (این حالت وقتی است که سرویس دهندهٔ محلی قبلاً آن نام را ترجمه و در یک فایل ذخیره کرده باشد) در غیر این صورت سرویس دهندهٔ محلی خودش یک تقاضا برای DNSسطح بالا ارسال میکند. این سرویس دهنده ، آدرسِ ماشینی را که میتواند برای ترجمهٔ نام مورد نظر مفید باشد به سرویس دهندهٔ محلی معرفی میکندسرویس دهندهٔ محلی مجدداً یک تقاضا به ماشین معرفی شده در مرحلهٔ قبل ارسال میکند در این حالت هم سرویس دهندهٔ نام میتواند در صورت یافتن آدرس IPمعادل با آن نام حوزه آنرا ترجمه کند و یا آنکه آدرس سرویس دهندهٔ سطح پایین تری را به او برگرداند

این روند ادامه می یابد تا DNS نهائی نام مورد نظر را به آدرس IPترجمه نماید

برای درک بهتر از روند کار به شکل (۱- ۶) دقت کنید


Query for address of
.com name server
Root
Name
Server
Root
Referral to .com
name server
Query for address of
microsoft.com name server
.com
Name
Server
Local
Name
Server
.com
.edu
Referral to microsoft.com
name server
Query for address of
www.microsoft.com
microsoft.com
Name
Server
.microsoft.com
IP Address
Resolver
Query
Response
Resolver

- در این مثال فرض شده است که یک برنامه کاربردی بافراخوانی "تابع تحلیلگر نام" تقاضای ترجمهٔ نام www.microsoft.com را می نماید. مراحلی که انجام میشود به شرح ذیل است:
- در مرحلهٔ اول برنامهٔ کاربردی با فراخوانی "تابع تحلیل نام" ، تقاضای ترجمه آدرسwww.microsoft.com را برای سرویس دهندهٔ محلی ارسال کرده و منتظر میماند
- در مرحلهٔ دوم ، سرویسدهندهٔ محلی از سرویس دهندهٔ Root(که حوزه های متفاوت راتفکیک میکند) آدرسِ ماشین یک DNSکه متولّی حوزه .comاست را سؤال میکند.
- در مرحلهٔ سوم ، آدرس سرویس دهندهٔ مربوط به حوزهٔ . comبرمی گردد
- درمرحلهٔ چهارم ، سرویس دهندهٔ محلی ، از ماشین معرفی شده در مرحلهٔ قبلی ، آدرس سرویس دهندهٔ مربوط به حوزه microsoft.comرا سؤال می نماید.
- درمرحلهٔ پنجم فهرستی از سرویس دهنده های DNSمربوط بهmicrosoft.comبرمیگردد
- در مرحلهٔ ششم ، سرویس دهندهٔ محلی تقاضای ترجمهٔ آدرس نمادین www.microsoft.com از DNSمتعلق به حوزه microsoft.comمیکند.
- در مرحلهٔ هفتم ، معادل آدرس IPنام www.microsoft.comبرمیگردد
- در مرحلهٔ هشتم ، آدرس IPخواسته شده در اختیار برنامه کاربردی قرار میگیرد

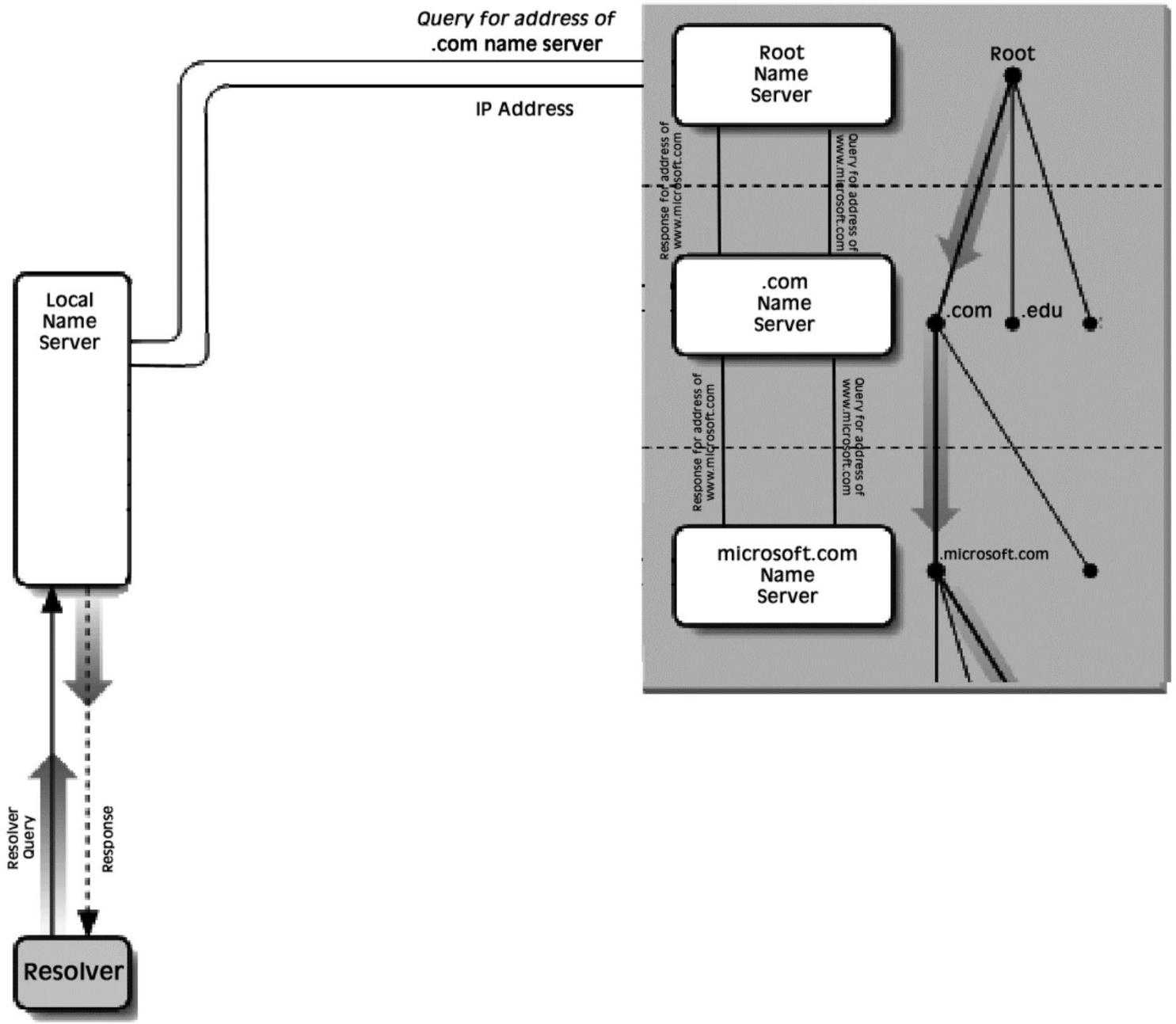
پرس و جوی بازگشتی

در این روش هر گاه برنامهای بخواهد آدرس IPمعادل یک نام مثل cs.yale.edu را بدست آورد بگونه ای که قبلاً اشاره شد "تابع سیستمی تحلیل نام" را فراخوانی میکند. این تابع یک ماشین را بعنوان سرویس دهندهٔ محلی از قبل می شناسد و بنابراین تقاضای تبدیل نام را به روش UDP برای آن ارسال کرده و منتظر جواب می ماند (پاسخ نهایی DNSطبیعتاً باید یک آدرس ۳۲ بیتی معادل آدرس IPیک ماشین باشد) دو حالت ممکن است اتفاق بیفتد:

- ممکن است در بانک اطلاعاتی مربوط به سرویس دهندهٔ محلی ، آدرس IP معادل با آن نام از قبل وجود داشته و بالطبع به سرعت مقدار معادل IPآن برمیگردد
- ممکن است در بانک اطلاعاتی سرویس دهندهٔ محلی ، معادل IPآن نام وجودنداشته باشد. مثلاً سرویس دهندهٔ محلی در بانک اطلاعاتی خودش معادل آدرس IPنام cs.mit.edeuرا نداشته و طبیعتاً نمی تواند آن را ترجمه کند. در چنین حالتی سرویس دهندهٔ محلی موظف است بدون آنکه به تقاضا دهنده خبر بدهد ، خودش رأساً به سرویس دهندهٔ سطح بالاتر تقاضای ترجمه آدرس بدهد دراین حالت هم dns سطح بالاتر به همین نحو ترجمه آدرس را پیگیری میکند یعنی اگر معادل IPآن نام را داشته باشد آنرا برمی گرداند و در غیر اینصورت خودش از سرویس دهندهٔ سطح پایین تر تقاضای ترجمهٔ آن نام را مینماید و این مراحل تکرار میشود.

در روش پرس و جوی بازگشتی ماشین سرویس دهندهٔ محلی این مراحل متوالی را نمیبیند و هیچ کاری جز ارسال تقاضای ترجمهٔ یک آدرس برعهده ندارد و پس از ارسال تقاضا برای سرویس دهندهٔ سطح بالا منتظر خواهد ماند. باز هم تکرار میکنیم ، روشی که DNS برای ترجمه آدرس بکار میبرد میتواند بدون اتصال UDP باشد که این کار به سرعت عمل ترجمهٔ آدرس می افزاید.

در روش پرسوجوی تکراری نسبت به روش پرسوجوی بازگشتی حجم عمدهٔ عملیات برعهدهٔ سرویس دهندهٔ DNS محلی است و مدیریت خطاها و پیگیری روند کار ساده تر خواهد بود و روش منطقی تری برای بکارگیری در شبکهٔ اینترنت محسوب میشود. روش پرسوجوی بازگشتی برای شبکه های کوچک کاربرد دارد.


Query for address of
.com name server
IP Address
Local
Name
Server
Root
Name
Server
.com
Name
Server
microsoft.com
Name
Server
Response for address of
www.microsoft.com
Query for address of
www.microsoft.com
Response for address of
www.microsoft.com
Query for address of
www.microsoft.com
Root
.com
.edu
.microsoft.com
Resolver
Query
Response
Resolver

پرس و جوهای معکوس

فرض کنید حالتی بوجود بیاید که یک سرویس دهندهٔ DNS، آدرس IP یک ماشین را بداند ولی نام نمادین معادل با آن را نداند. بعنوان مثال DNS مایل است بداند که چه نامی در شبکهٔ اینترنت معادل با میباشد. در چنین حالتی مسئله کمی حادّتر به نظر میرسد ، چرا که۱۹۵.۱۳.۴۲.۷ برای ترجمهٔ نام های نمادین ، چون این نام ها دارای حوزه و زیرحوزه هستند ، تحلیل آدرسها ساده است ولی ترجمهٔ آدرس IP به معادل نام حوزه ، از چنین روابطی تبعیت نمیکند؛ بعبارت بهتر هیچ ارتباط مستقیم و متناظری بین آدرسهای IPو اسامی انتخاب شده در اینترنت وجود ندارد برای یافتن نامهای متناظر با یک آدرس IPباید یک جستجوی کامل و در عین حال وقت گیر انجام بشود

روش کاربدین صورت است که سرویس دهندهٔ محلی یک تقاضا برای DNS متناظر با شبکه ای که مشخصهٔ آن در آدرس IP، مشخص شده ، ارسال می کند بعنوان مثال آدرس IPشبکهای را ۱۳۸.۱۴.۷.۱۳ در نظر بگیرید آدرس کلاسB و مشخصهٔ آن ۱۳۸.۱۴.۰.۰ است. زمانی که موسّسه ای یک کلاس IPثبت میدهد یک سرویس دهندهٔ DNS متناظر با شبکهٔ خود ایجاد کرده و آنرا نیز معرفی میکند.

سرویس دهندهٔ محلی بایستی آدرس DNSمتناظر با شبکهٔ ۱۳۸.۱۴.۰.۰ را پیدا کرده و سپس برای آن یک تقاضا ارسال کند. DNS مربوط به این شبکه ، براساس زیرشبکه هایی که دارد این سؤال را از طریق سرویس دهنده های متناظر با هر زیرشبکه پیگیری میکند.(چون هر زیر شبکه یک سرویس دهندهٔ DNSمخصوص به خود دارد) نهایتاً یک نامِ نمادین حوزه معادل با آن آدرس IPبرخواهد گشت.

پروتکل پیکربندی پویا میزبان یاDHCP

- کاربرد DHCP SERVER

زمانی که شما می خواهید به یک شبکه محلی منتقل شوید باید آدرسIP شبکه محلی در یک کلاس باشد یعنی باید IP Address و Subnet mask متناسب با رایانه های دیگر شبکه تنظیم شده.

بخواهیم با استفاده از از روتر(مانند مودم ADSL) به شبکه دیگر مانند اینترنت منتقل شوید باید Defult gatway که همان IP روتر هست را تنظیم نمایید.

انجام تنظیمات فوق به صورت دستی به وقت و زمان زیادی نیاز دارد لازم به ذکر است تنظیمات فوق برای تک تک رایانه های موجود در شبکه باید انجام بگیرد که کار طاقت فرسایی است.

در صورت کوچکترین اشتباه امکان تداخل، عدم اتصال به شبکه به وجود خواهد آمد.
مشکلات مطرح شده از سرویس DHCP می توان استفاده نمود.

سرویس DHCP پروتکلی است که با استفاده از آن شما می توانید ججدوده ای از آدرس های IP را انتخاب و سپس این پروتکل به صورت ماتمرکز پیکره بندی آدرس IP و دیگر تنظیمات TCP/IP میزبان های شبکه به صورت پویا و خودکار انجام شده و کاربران شبکه را از پیکربندی دستی آدرس های Ip بی نیاز می نماید.

این قابلیت برای کامپیوتر های بی سیم همراه که به طور موردی درشبکه حضور دراند بسیار حیاتی است. کاربران هیچ علاقه ای ندارند با جابه جا شدن در شبکه یا تعویض ISP خود تنظیمات کامپیوتر خود را به صورت دستی وارد کنند. مسئولین شبکه هم تمایلی به اختصاص آدرس های IP به کامپیوترهایی که گاه هستند و گاه نیستند ندارند.

معایب استفاده از DHCP :

۱- مشکل امنیتی(رایانه غیر مجاز هم می تواند ازسرور درخواست IP داشته باشد)

۲- در صورت خرابی سرویس DHCP تخصیص IP به کلاینت ها با شکست مواجه می شود.

۳- افزایش ترافیک شبکه به خاطر ارتباط کلاینت ها DHCP اتقاف خواهد افتاد.

مزایای استفاده از DHCP:

از آنجایی که DHCP به صورت متمرکز اختصاصIP Address را مدیریت و کنترل می نماید از ایجاد Confilice در IP یعنی اختصاص دو IP مشابه به صورت اشتباه جلوگیری می کند.

تشرح عملکردDHCP

وقتی یک کاربر رایانه خود را راه اندازی می نماید سیستم عامل آن بعد از بالا آمدن در خواست IP می کند و آن را از سرویس دهنده ی DHCP دریافت می کند. به طوری که این درخواست به ترتیب در طی چهار مرحله تکمیل می شود.

:DHCP Discover

در این مرحله کلاینت پیغام DHCP Discover خوذ را جهت دریافت IP در شبکه broadcast می کند و به آدرس ۲۵۵.۲۵۵.۲۵۵.۲۵۵ ارسال می کند.در این حالت IP کلاینت به صورت 0.0.0.0 خواهد بود.

پیشنهاد IP از طرف سرور DHCP یا DHCP Offer

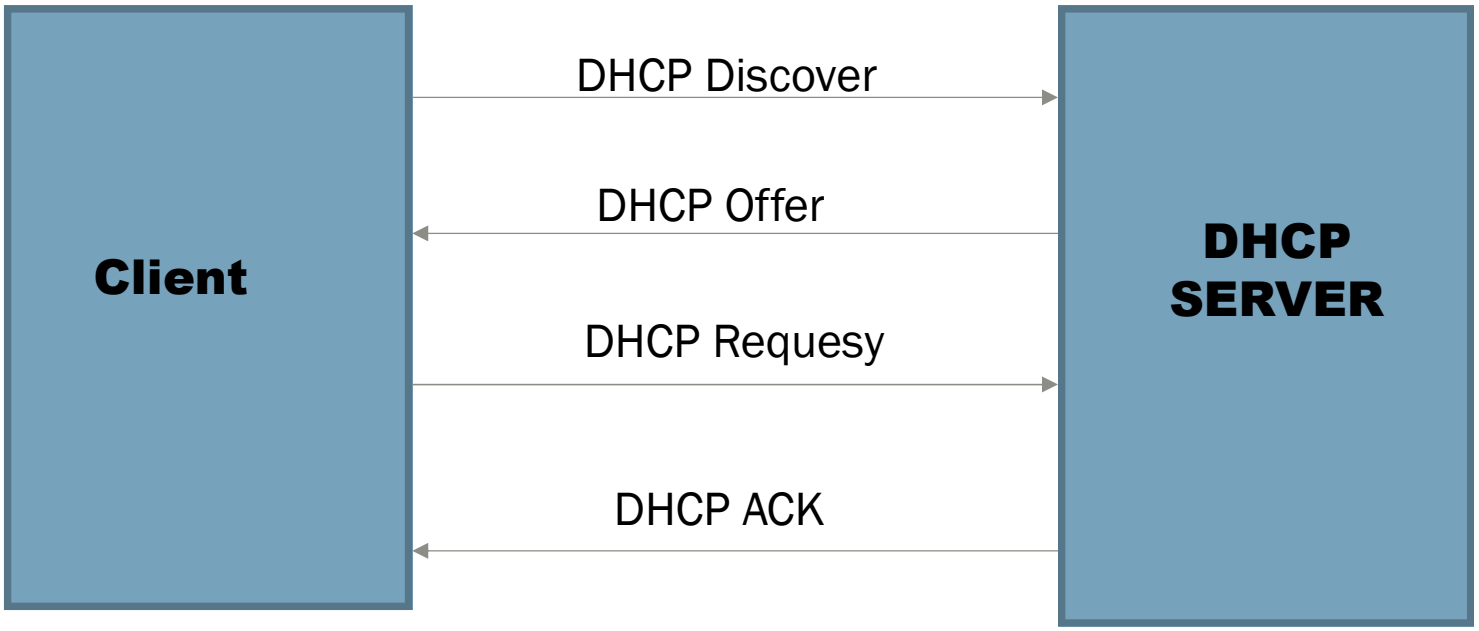
در این مرحله تمام سرویس دهنده های DHCP از محدوده آدرس IP تعریف شده بر روی خود یک IP انتخاب نموده و به همراه مدت زمانی که قرار است آن IP را در اختیار کلاینت قرار دهد (پیغام DHCP Offer) و آن را به کلاینت ارسال می نماید.

در خواست برای IP پیشنهادی یا DHCP Request:

کلاینت درخواست کننده پس از دریافت Ipهای پیشنهادی ،اولین IP پیشنهادی را انتخاب نموده و آن را توسط یک Packet در شبکه Broadcast می کند و در آن Packet آدرس سرویس دهندهDHCP را که پیشنهاد او قبول شده است مشخص می نماید.

تایید درخواست IP با پیغام DHCP ACK:

پس از آن که سرویس گیرنده به DHCP Server که پیشنهاد اوقبول شده DHCP Request را فرستاد در صورتیکه هنوز IP پیشنهاد شده در رنج IP های سرویس دهنده DHCP وجود داشته باشد و توسط Admin حذف نشده باشد DHCP Srever تایید خود را مبنی بر اختصاص IP به سرویس گیرنده اعلام می کند.و اگر IP توسط Admin از محدوده مربوطه حذف شده باشد DHCP Server به کلاینت در خواست کننده DHCP Nack را ارسال می کند، client مجبور می شود تمام مراحل را دوباره طی کند.


Client
DHCP Discover
DHCP Offer
DHCP Requesy
DHCP ACK
DHCP
SERVER