

RFC 1034 به معرفی سیستم حوزه نام CDNS و مفاهیم آن می پردازد و از بعضی

مطالب ذکر شده در RFC 1035 که مربوط به پیاده سازی است صرف نظر می کند.

مجموعه توابع و انواع داده ها در DNS ، تشکیل یک پروتکل رسمی و موثق می دهد.

که شامل پرس و جویهای استاندارد، پاسخهای آنها و قالب کلاسهای اینترنت می باشد.

با این وجود، سیستم حوزه نام عملاً توسعه پذیر باقی گذاشته شده است. به طور مثال

متنوباً پیشنهادهای جدیدی در مورد پیاده سازی، انواع داده، انواع پرس وجود کلاستها و

توابع ارائه می شوند.

بنابراین تا زمانی که انتظار می رود پروتکل رسمی بدون تغییر بماند، هرگونه فعالیت

آزمایشی باید در امتداد این پروتکل انجام شود.

همچنین خواننده باید آگاه باشد که مثالهای ذکر شده حتماً کامل و اجرایی نیستند و

صرفاً جنبه آموزشی دارند.

توزیع این RFC هیچگونه محدودیتی ندارد.

معرفی

این RFC به معرفی قالب نامه های حوزه، استفاده از آنها در پست الکترونیکی و

پروتکلها و سرورهای ارائه شده برای پیاده سازی سرویس حوزه نام می پردازد.

تاریخچه نامهای حوزه:

انگیزه توسعه سیستم حوزه نا، با گسترش اینترنت آغاز شد.

- در ابتدا پیدا کردن آدرس ماشین میزبان از طریق فایل hosts.txt انجام می شد و

مسئولیت آن بر عهده NIC بود. این فایل به طریق FTP توسط همه میزبانها ارسال می

شد. اما پهنای باند مصرف شده در شبکه برای ارسال این فایل با مجوز تعداد میزبانها متناسب است و حتی هنگامی که چند سطح مختلف از FTP استفاده می شود. این پهنای باند بسیار قابل ملاحظه است. با افزایش تعداد میزبانها، این روش عملاً کارایی خود را از دست می دهد.

- مسئله بعد تغییر در خصوصیات اینترنت بود. میزبانهای با زمان اشتراکی دوران Arpanet حال جای خود را به شبکه هایی از ایستگاههای کاری داده بودند. همچنین بسیاری از سازمانها را نام و آدرسهای خود استفاده می نمودند ولی مجبور بودند صبر کنند. NTC hasts.txt را به روز کند تا تغییراتشان اعمال گردد.

- از سوی دیگر کاربردهای اینترنت مرتباً پیچیده تر می شد و نیاز به یک سیستم نام چند منظوره بیش از پیش به چشم می خورد.

در نتیجه این مشکلات ایده های جدیدی درباره قالب و مدیریت نامها مطرح شد که در اغلب آنها ساختار سلسله مراتبی پیشنهاد شده بود.

در این روش از کاراکتر « . » به عنوان جدا کننده بین سطوح مختلف استفاده می شود.

یک طراحی از این سیستم با استفاده از پایگاه داده توزیع شده و منابع تعمیم یافته در RFC 882 شرح داده شده است.

واژه « domain » (حوزه) در خیلی متون به چشم می خورد که در اغلب موارد برای ارجاع به نامی که با علامت « . » قسمتهای آن از همه جدای گردند، استفاده می شود ولی

ارتباطی به DNC ندارد.

اهداف طراحی DNS:

- هدف اولیه ایجاد یک روش نامگذاری یکسان برای ارجاع به منابع می باشد. برای اجتناب از بعضی مشکلات نامها باید شامل آدرسها، مسیرها، و یا اطلاعات مشابه دیگر باشند.

- حجم پایگاه داده و کثرت به روز رسانی آن باعث می شود که از سیستمهای توزیع شده استفاده شود. که برای کارایی بهتر می توان از کش کردن محلی نیز استفاده نمود. جمع کردن کل پایگاه به طور متمرکز بسیار پرهزینه و مشکل می باشد و باید اجتناب گردد.

- هنگام سبک سنگین کردن بینا فرینه به دست آوردن داده ها سرعت به روز سانی و دقت کشی ها برتری با داده ها می باشد.

- هزینه پیاده سازی ما را مجبور می کند که یک سیستم چند منظوره ایجاد کنیم که برای بازیابی آدرسهای میزبانها، داده های صندوق های پستی و موارد دیگر قابل استفاده باشد.

- برای اینکه این سیستم برای شبکه های مختلف قابل استفاده باشد. باید قابلیت استفاده از یک سیستم نامگذاری برای پروتکل های مختلف را فراهم کنیم. برای مثال آدرسهای میزبان در پروتکل های مختلف، متفاوتند ولی همه آنها مفهوم آدرس را دارند. DNS همه داده ها را در کلاس بر چسب (tag) می زند و اجازه استفاده موازی از قالبهای مختلف را می دهد.

- از دیگر مطالبات ما استقلال تراکتهای سرور نام از سیستم حامل آنها می باشد بعضی سیستمها ممکن است برای پرس و جو و پاسخ از دیتاگرام استفاده کنند ولی بعضی سیستمها استفاده انحصاری از مدارهای مجازی را در نظر داشته باشند.

- سیستم باید در محدوده وسیعی از قابلیت‌های میزبان مفید باشد. هم کامپیوترهای شخصی و هم سیستم‌های بسیار بزرگ باید بتوانند از سیستم ولو از راه‌های متفاوت استفاده نمایند.

- مفروضاتی راجع به استفاده از سیستم:

- سازمان بندی سیستم حوزه از بعضی فرخها راجع به احتیاجات و استفاده کاربران نشأت می گیرد و طوری طراحی شده است که از بعضی مشکلات پیچیده پایگاه داده ها اجتناب شود.

این مفروضات به شرح زیر است:

- حجم پایگاه داده کلی به تعداد میزبانهایی که از سیستم استفاده می کنند بستگی دارد. ولی در نهایت به تعداد کاربران آن میزبانها نیز وابستگی پیدا می کند مثلاً کاربران صندوق پستی و بقیه اطلاعات آن میزبان
- اکثر اطلاعات داخل سیستم به کندی تغییر می یابند ولی سیستم باید قادر باشد تا با تغییرات سریع نیز سرو کار داشته باشد.
- سرویس گیرنده های سرویس حوزه باید قادر باشند تا سرورهای نام مورد اعتماد خود را که ترجیح می دهند از آنها سرویس بگیرند شناسایی کنند و خدمات مورد نیاز خود را از این سرورها اخذ کنند.
- دسترسی به اطلاعات حیاتی از به روز رسانی فوری و یا گارانتی سازگاری است. به روز رسانی اجازه رد شدن اطلاعات از طریق کاربران به سیستم حوزه را می دهد.

هنگامی که به روز رسانی به دلیل نقص فنی در دسترس نیست، داده های فعلی به عنوان قدیمی شناخته می شوند و تلاش مستمر در جهت به روز رسانی آن باید انجام گیرد.

مدل کلی به این صورت است که کپی ها با یک مدت زمان برای تازه سازی توزیع می شوند توزیع کننده این مدت زمان را تنظیم می کند و گیرنده توزیع مسئولیت تازه سازی را به عهده دارد. در شرایط خاصی می توان مدت زمان بسیار کمی را تنظیم کرد و یا از توزیع کپی ها جلوگیری به عمل آورد.

- در هر سیستم با پایگاه داده توزیع شده، ممکن است یک سرویس دهنده نام و یک پرس وجو ارائه شود که فقط توسط یک سرور دیگر بتواند پاسخ داده شود.

دو رویکرد کلی در برخورد با این مسأله «بازگشتی» و تکرار شونده می باشند.

در روش اول (بازگشتی) سرور اول پرس وجو سرویس گیرنده را از طریق یک سرور دیگر تعقیب می کند. در روش «تکرار شونده» سرور، سرویس گیرنده را به یک سرور دیگر ارجاع می دهد و اجازه می دهد که سرویس گیرنده خود پرس وجو را دنبال کند.

هر دو رویکرد دارای مزایا و معایبی می باشند ولی روش تکرار شونده برای دسترسی از طریق دیتاگرام ترجیح دارد. سیستم حوزه نیازمند پیاده سازی رویکرد تکرار شونده است ولی اجازه رویکرد بازگشتی را نیز می دهد. سیستم حوزه فرض می کند که همه داده ها از فایلها اصلی (Master) سرچشمه می گیرند و در میزبانهای مختلف پراکنده شده اند.

master files فایلهای متنی هستند که توسط یک سرویس دهنده نام محلی خوانده می شوند و بنابراین از طریق سرویس دهنده های نام برای کاربران سیستم حوزه در دسترس می باشند.

برنامه های کاربران از طریق برنامه هایی که حل کننده (Resolver) خوانده می شوند به سرویس دهنده های نام دسترسی پیدا می کنند.

قالب استاندارد Master فایلها به آنها اجازه می دهد که بین میزبانهای مختلف از طریق Mail,FTP و یا روشهای دیگر مبادله شوند.

این امکان به خصوص هنگامی مفید است که یک سازمان یک نام حوزه می خواهد ولی نمی خواهد که یک سرویس دهنده نام راه اندازی کند. سازمان می تواند مستر فایلها را با یک ویرایشگر متن به طور محلی تهیه کند و سپس آنها را به یک میزبان خارجی که دارای سرویس دهنده نام می باشد انتقال دهد. و سپس با سرپرست سرویس دهنده نام قرار بگذارد که فایلها را بار گذاری کنید.

هر کدام از سرویس دهنده های نام و Resolver های میزبانها توسط یک سرپرست سیستم پیکر بندی می شوند.

برای یک سرویس دهنده نام اطلاعات پیکر بندی شامل : هویت مستر فایلها محلی و دستورالعملهایی برای بارگذاری مستر فایلها غیر محلی می باشند. سرویس دهنده های نام این مستر فایلها را کپی کرده و یا آنها را کپی می کنند.

برای Resolver ها، اطلاعات پیکر بندی سرویس دهنده نامی را که منبع اولیه اطلاعات می باشد را مشخص می کند.

سیستم حوزه روشهایی برای دسترسی به داده ها و ارجاع به دیگر سرویس دهنده های نام را تعریف می کند. همچنین تعریف روشهایی برای کش کردن اطلاعات بازیابی شده

و تازه سازی دوره ای آنها که ز طرف مدیر سیستم تعریف شده است، از دیگر وظایف سیستم حوزه می باشد.

مدیران سیستم موارد زیر را فراهم می کنند:

- تعریف مرزهای بین مناطق و ناحیه های مختلف (Zone)

- مستر فایل های داده ها

- به روز رسانی مستر فایلها

- بیان سیاستهای تازه سازی مطلوب

سیستم حوزه موارد زیر را فراهم می کند:

- قالب استاندارد برای داده های منبع

- روشهای استاندارد برای پرس و جو از پایگاه داده

- روشهای استاندارد برای سرویس دهنده های نام برای تازه سازی اطلاعات محلی از

طریق سرویس دهنده های نام خارجی

عناصر یک DNA :

یک DNA سه قطعه اصلی دارد:

- فضای حوزه نام و پرونده های منابع که مشخصات درخت فضای نام ساخت یافته و

داده های مرتبط با نامها هستند.

به طور مفهومی هر گره و برگ از درخت فضای نام یک سری اطلاعات را نامگذاری

می کند. و پرس و جوها تلاش هایی برای استخراج اطلاعات خاصی از این سری ها

می باشند.

یک پرس و جو یک نوع از اطلاعات منبع را که مطلوب است تشریح می کند.

برای مثال اینترنت بعضی از نامهای حوزه خود را برای تشخیص دادن میزبانها به کار می برد پرس و جو برای منابع آدرس آرس میزبان را برمی گرداند.

- سرویس دهنده های نام، برنامه های مسروری هستند که اطلاعاتی درباره ساختار درخت حوزه و اطلاعات مجموعه ها را نگه می دارند.

یک سرویس دهنده نام همگن است ساختار و اطلاعات هر قسمت از درخت حوزه را کش کند ولی به طور معمول هر سرویس دهنده نام اطلاعات کاملی از زیر مجموعه فضای حوزه و اشاره گرهایی به دیگر سرویس دهنده های نام که می تواند منتهی به اطلاعاتی از درخت حوزه شود دارد.

- Resolverها برنامه هایی هستند که بنا به درخواست مشتری، اطلاعات را از سرویس دهنده های نام استخراج می کنند. Resolverها حداقل باید قادر به دسترسی به یک سرویس دهنده نام باشند تا با استفاده از آن بتوانند به یک پرس و جو مستقیماً جواب داده و یا از طریق ارجاع به سرویس دهنده های نام دیگر به نتیجه دلخواه برسند.

به طور معمول یک Resolver، یک رویه سیستم است که به طور مستقیم به برنامه های کاربران دسترسی دارد. بنابراین بین Resolver و برنامه های کاربران احتیاج به پروتکل های واسط نمی باشد.

یک سیستم حوزه، دارای سه لایه یا دیدگاه می باشد.



- از نقطه نظر کاربرد، سیستم حوزه از طریق یک روش ساده و یا یک درخواست سیستم عامل قابل دسترسی است. فضای حوزه شامل یک درخت ساده است و کاربر می تواند از هر قسمتی از درخت اطلاعات را درخواست کند.

- از دیدگاه Resolver، سیستم حوزه از تعداد نامشخص سرویس دهنده نام تشکیل شده که هر کدام از آنها یک یا بیشتر تکه از کل اطلاعات درخت حوزه را دارد ولی Resolver هر کدام از این پایگاه های داده را به طور استاتیک می بیند.

- از دیدگاه سرویس دهنده نام، سیستم حوزه شامل مجموعه های جداگانه ای از اطلاعات محلی که ناحیه خوانده می شوند می باشد. سرویس دهنده نام کپی های محلی از بعضی از این نواحی دارد.

- سرویس دهنده نام باید به طور مرتب و دوره ای با استفاده از کپی های اصلی محلی و یا استفاده از سرویس دهنده های نام خارجی نواحی خود را تازه سازی کند.

همچنین باید به طور همزمان پرس و جوهای رسیده از Resolverها را پردازش کند.

هنگامی که کارایی برای ما اهمیت پیدا می کند، پیاده سازی ممکن است دو تا از این توابع را ترکیب کند. برای مثال یک Resolver بر روی همان ماشین سرویس دهنده نام ممکن است به طور اشتراکی از پایگاه داده ها استفاده کنند. نواحی توسط سرویس دهنده نام و

کپی کردن توسط Resolver مدیریت شود.

- فضای حوزه نام و پرونده منابع:

- مشخصات و واژه شناسی فضای نام:

فضای نام حوزه یک ساختار درختی است. هر گره و برگ در این درخت نشان دهنده یک مجموعه منبع می باشد (ممکن است خالی باشد)

سیستم حوزه هیچ تفاوتی بین استفاده از نودهای داخلی و برگ نمی گذارد و ما نیز از واژه گره برای ارجاع به هر دو استفاده می کنیم.

هر گره یک برچسب دارد که طول آن از ۰ تا ۶۳ متغیر است گره های برادر ممکن است برچسبهای برابر نداشته باشند. همچنان که دو گره که برادر نیستند ممکن است برچسب برابر داشته باشند. برای ریشه از برچسب صفر استفاده می شود.

نام حوزه یک گره لیست برچسبهای مسیر از ریشه تا آن گره می باشد. به طور قراردادی برچسبهایی که تشکیل یک نام حوزه می دهند از چپ به راست خوانده و نوشته می شوند.

برنامه هایی که نامهای حوزه را پردازش می کنند باید آنها را به صورت دنباله ای از برچسبها نمایش دهند.

چون تمامی نامهای حوزه به اوت ختم می شوند که رشته صفر برای برچسب دارد. این نمایش های داخلی می توانند از یک بایت صفر در انتهای نام حوزه استفاده کنند.

هنگامی که یک کاربر احتیاج به تایپ یک نام حوزه دارد، طول هر برچسب حذف می شود و برچسبها با نقطه از یکدیگر جدا می گردند. از آنجا که یک نام حوزه کامل با برچسب ریشه خاتمه می یابد، به یک فرم منتهی می شود که با نقطه پایان می پذیرد.

ما از این خاصیت برای تمیز دادن این موارد استفاده می کنیم:

- یک رشته کاراکتر که نشان دهنده یک نام حوزه کامل است مانند Poneria. Isi. Edu

- یک رشته کاراکتر که برجسب آغازین از یک نام حوزه را که ناکامل است نشان دهد و توسط یک نرم افزار باید کامل شود. برای مثال Poneria در مثال ذکر شده قبل.

- نامهای نسبی یا نسبت به یک نام مشهور قرار دارند و یا نسبت به یک لیست از نامهای حوزه که بری جستجو استفاده می شود.

هر حوزه Cdomain توسط یک نام حوزه مشخص می شود و شامل آن قسمتی از فضای نام حوزه است که در زیر نام حوزه ای که Domain را مشخص می کند قرار دارد.

یک حوزه، یک زیر حوزه از یک حوزه دیگر است اگر با آن حوزه سازگار باشد. ای ارتباط به این صورت آزمایش می شود که اگر نام زیر حوزه به قسمتی از نام حوزه ختم شود، آنگاه زیر حوزه خواهد بود. برای مثال A.B.C.D زیر حوزه B.C.D و C.D می باشد.

- راهنمای مدیران سیستم:

به عنوان یک سیاست، مشخصات فنی DNA تحت نظر یک ساختار درختی به خصوص نمی باشد، هدف این بوده که تا حد ممکن عمومی باشد تا بتوان با آن کاربردهای دلخواه را پیاده سازی کرد.

به طور اخص سیستم طوری طراحی شده است که فضای نام مجبور نباشد در کنارمرزهای شبکه سازمان دهی شود.

راهنماهای فنی برای استفاده:

قبل از اینکه DNA برای نگهداری اطلاعات راجع به نامگذاری مورد استفاده قرار گیرد، دو شرط زیر باید برقرار باشد:

۱- یک قرارداد برای نگاشت بین نامهای شئی و نامهای حوزه که تشریح می کند چگونه اطلاعات یک شئی مورد دسترسی قرار می گیرد.

۲- قالب داده‌ها برای شتريخ شئی

این قوانین می توانند ساده و یا بسیار پیچیده باشند. در اغلب موارد طراحان بای قالب و فرضتهای موجود را به حساب آورند و به طریقی سیستم را توسعه دهند که با قالبهای قبلی سازگار باشد.

برای میزبانها، نگاشت، بستگی به سینتکس موجود برای نامهای میزبان دارد. که زیر مجموعه از نمایشهای متنی موجود برای نامهای حوزه می باشد.

یک کاربرد ساده، علاقه‌ای به تعریف این قوانین ندارد ولی باید درک کند که این قوانین معمولاً نتیجه سازش بین سازگاری رو به بالا و استفاده‌های قدیمی می باشد.