

توی این قسمت میخوایم انواع آدرس ها در شبکه رو توضیح بدیم.

خب اول لازمه بدونید که آدرسینگ از لایه ۲ شروع میشه و توی لایه ۱ همونطور که احتمالا میدونید آدرسینگ وجود نداره. توی این بخش بیشتر تمرکز ما روی مک آدرس ، آی پی آدرس و پورت نامبر هاست یکمی هم راجع به url ها صحبت میکنیم.

خب اولین لایه ای که در اون از آدرس ها استفاده میشه لایه ۲ هستش
آدرس ها در لایه ۲ اسمش مک آدرس یا آدرس فیزیکی. چرا فیزیکی؟ چون به صورت فیزیکی روی کارت شبکه ثبت میشه و هرگز قابل تغییر نیست.
مک آدرس، یک آدرس ۴۸ بیتی هستش که به ۲ قسمت ۲۴ بیتی تقسیم میشه. یعنی چی؟ یعنی برای ایجاد مک آدرس ۴۸ تا صفر و یک در کنار هم قرار میگیره. ما این آدرس را ۴ تا ۴ تا تقسیم میکنیم و برای فهم بهتر به شکل هگزادسیمال در می آوریم.
مک آدرس در شکل هگزادسیمال از ۱۲ کاراکتر تشکیل میشه که ۲ تا ۲ تا با هم جفت می شوند و داخل ۲ تا ۲ نقطه قرار میگیرند مثل شکل زیر:

1a:12:8e:26:8b:19

این خیلی مهمه که ما بدونیم ماشین به این آدرس به عنوان یک بلوک ۴۸ بیتی نگاه میکنه و تقسیم بندی هایی که ما انجام میدهیم و تبدیل به اعداد میکنیم فقط برای فهم خودمونه و برای کامپیوتر اهمیتی نداره. حالا چطور ما این اعداد باینری رو به هگزادسیمال تبدیل میکنیم؟ توی فرم هگزادسیمال هر آیکون بین ۰ تا ۱۵ میتونه مقدار بگیره یعنی جمعا ۱۶ تا

خب ما توی زبان کامپیوتر اگر بنویسیم ۱۰ خونده میشه ۱ و ۰ پس ما نمیتونیم مقدار ۱۰ رو بدیم.
توی فرم هگزادسیمال مقادیر ۰ تا ۹ به شکل خودشون نوشته میشن اما از ۱۰ تا ۱۵ با حروف انگلیسی مشخص میشن

۱۰ f ۱۱ e ۱۲ d ۱۳ c ۱۴ b ۱۵ a

خب برگردیم سراغ تبدیل؛ گفتیم مک آدرس ۴۸ بینه که ما ۴ تا ۴ تا جدا میکنیم هر کدوم از این ۴ تا یک مقداری دارن و این مقادیر توان های عدد ۲ هستن چرا؟ چون کامپیوتر یک ماشین ۰ و ۱ هستش پس مقادیر لگاریتم عدد ۲ هستن.

خب این ۴ تا از راست به چپ به ترتیب ۱، ۲، ۴ و ۸ هستن

یعنی ۲ به توان ۰، ۲ به توان ۱، ۲ به توان ۲ و ۲ به توان ۳

خب حالا اینا چجوری تبدیل به ۰ و یک میشن؟

وقتی ما میگیریم e یعنی ۱۴ باید این عدد ۱۴ رو با اون ۴ تا عدد بسازیم :

$$1 = 8$$

$$1 = 4$$

$$1 = 2$$

$$0 = 1$$

$$\text{پس } 14 = 8 + 4 + 2 + 0 \text{ میشه}$$

یا برعکس اگر مقدار باینری ۱۱۰۱ رو داشته باشیم میشه :

$$13 = 8 + 4 + 0 + 1$$

پس 2b اگر بخوایم به شکل باینری برگردونیم میدونیم که ۲ میشه :

$$0010$$

و b که نماد عدد ۱۱ است میشه :

$$1011$$

خب اینکه مک آدرس به چه دردی میخوره رو قبلا خیلی راجع بهش صحبت کردیم و فهمیدیم کلا مسیریابی

لایه دویی داخل سوئیچ ها با مک کار میکنه و جداول مک توی سوئیچ ها مسیر هارو تعیین میکنه.

چیزی که خیلی مهمه به خاطر داشته باشیم اینه که آدرس مک خریدنی و شرکت های تولید کننده تجهیزات

شبکه میرن OSI و ۲۴ بیت اول رو از OSI خریداری میکنن و ۲۴ بیت دوم رو خودشون تولید میکنن و

دستگاه های مختلف تولید میکنن و حتی بستر های مجازی سازی که ما استفاده میکنیم و توی اون میتونیم

مک آدرس جنریت کنیم هم اون ۲۴ بیت اول رو خریدن و ثابت و ما فقط ۲۴ بیت دوم رو تغییر میدیم.

یه نکته دیگه که مهمه بدونید اینه که مک آدرس یک routed protocol نیست. یعنی چی؟ یعنی قابلیت

مسیریابی شدن رو ندارن. علتش هم اینه که همیشه گروهی مک آدرس هارو مسیر یابی کرد.

خب مشخصه که شرکت وقتی مثلا ۱۰۰,۰۰۰ تا دستگاه با مک آدرس های پشت هم تولید میکنه تضمینی

نیست که این دستگاه ها همه توی یک منطقه باشن که بشه همه ی اون آدرس هارو به اون سمت هدایت کرد

و آدرس های پشت هم در دنیا پخش میشن.

مهمترین چیز اینه که همیشه با شنیدن مک آدرس این جمله تو ذهنت روشن شه:

مک آدرس یک آدرس فیزیکی غیر قابل تغییر ۴۸ بیتی هست که در شکل هگزادسیمال به ۱۲ قسمت ۴ بیتی

تقسیم میشه و هرکدوم از ۱۲ قسمت میتونه بین ۰ تا ۱۵ یا همون f مقدار بگیره

چیزی که خیلی مهمه بدونیم اینه که در اول و آخر هر ارتباطی مک ها هستن که بسته هارو جابجا میکنن و اگر لایه ۲ ما درست کار نکنه هیچ ارتباطی نخواهیم داشت.

آدرس بعدی که میخوایم راجع بهش حرف بزنیم ip آدرس هستش. تعریف سرراست آی پی مثل تعریفی که از مک کردیم اینه که :

آی پی آدرس یک ادرس ۳۲ بیتی هست که به ۴ قسمت ۸ بیتی تقسیم شده که به این قسمت ها اوکند میگیریم. آی پی به ۲ قسمت host id و net id تقسیم میشه.

حالا بریم ببینیم این یعنی چی:

این ۸ بیتی ها مثل همون ۴ بیتی ها هستن با این تفاوت که از ۲ به توان ۰ تا ۲ به توان ۷ هستن که میشه ترتیب زیر:

$$۲ \text{ به توان } ۰ = ۱$$

$$۲ \text{ به توان } ۱ = ۲$$

$$۲ \text{ به توان } ۲ = ۴$$

$$۲ \text{ به توان } ۳ = ۸$$

$$۲ \text{ به توان } ۴ = ۱۶$$

$$۲ \text{ به توان } ۵ = ۳۲$$

$$۲ \text{ به توان } ۶ = ۶۴$$

$$۲ \text{ به توان } ۷ = ۱۲۸$$

خب پس هر کدوم از این ۴ قسمت میتونه بین ۰ تا ۲۵۵ مقدار بگیره .

توی آی پی ما فرم هگزادسیمال نداریم بلکه دسیمال داریم. تبدیل باینری به دسیمال هم توی آی پی مثل مک هستش. یه مثال ببینیم :

۱۱۲

۱۲۸ - ۱۱۲ همیشه پس میشه ۰

۶۴ - ۱۱۲ همیشه پس میشه ۱ و حاصل میشه ۴۸

۳۲ - ۴۸ همیشه پس میشه ۱ و حاصل میشه ۱۶

۱۶ - ۱۶ همیشه پس میشه ۱ و حاصل میشه ۰

و چون حاصل صفر شده مقادیر دیگه هم ۰ میشن

پس ۱۱۲ به باینری میشه:

۰۱۱۱۰۰۰۰

یا

۱۰۱۰۱۰۱۰ به دسیمال میشه

$$170 = 128 + 0 + 32 + 0 + 8 + 0 + 2 + 0$$

خب این از قسمت ۳۲ بیتی و ۴ قسمت ۸ بیتی جمله بالا. حالا بریم سراغ بحث اصلی و مهم آی پی host id و net id .

برای درک بهتر این موضوع باید به این مسئله توجه کنید که آی پی یک routed protocol هستش و قابلیت مسیریابی داره و علتش هم اینه که قابلیت مسیریابی گروهی داره چطوری؟

ببینید آی پی آدرس هارو در دنیا یک شرکتی به نام iana یا Internet Assigned Numbers Authority توزیع میکنه. یعنی چی یعنی کل آی پی های قابل استفاده دنیارو (جلوتر میفهمیم که یسری آی پی ها داخلی اند و یکسری برای اهداف دیگه اند و همیشه فروختشون) به چندین دسته تقسیم کرده و هر دسته رو به یک کشوری فروخته. خب حالا که مثلا ۳۰۰۰ تا آدرس پشت سر هم رو فروخته به ایران نیازی نیست ۳۰۰۰ تا مسیر متفاوت تعریف کنه کافیه به هر ۳۰۰۰ تا اشاره کنه و به سمت ایران هدایتشون کنه چیزی که فهمیدیم توی مک آدرس نداریم.

خب این چه کمکی میکنه؟ فرض کنید یک بسته از گوگل کالیفرنیا میخواد بیاد برای شما؛ آخرین روتر آمریکا آدرس شمارو بررسی میکنه و میبینه آدرس مربوط به ایرانه و تحویل ایران میده. حالا توی ایران برای اینکه هدایت شه به سمت شما نیاز نیست تمام مسیر ها بررسی بشن بلکه زیرساخت چک میکنه که این دسته آدرس رو به چه سرویس دهنده ای فروخته و تحویل اون میده چون اون مثلا ۳۰۰۰ تارو خرد کرده به ۶ دسته ۵۰۰ تایی پشت سر هم و فروخته به ۶ تا سرویس دهنده. مثلا میده به شاتل حالا شاتل دوباره ۵۰۰ تای خودش رو مثلا کرده ۵ تا دسته ۱۰۰ تایی پشت سر هم و کشور رو ۵ تکه کرده به نماینده هاش آدرس داده. هر نماینده ۱۰۰ تای خودش رو مثلا به ۱۰ تا آدرس پشت سر هم تقسیم کرده و داده به استان ها و استان ها از ۱۰ تا به هر شهر یدونه دادن پس برای مسیریابی بسته ۵ بار ۵ گروه چک میشن بجای اینکه ۳۰۰۰ تا آدرس چک بشه .

همه ی این هارو گفتم که بگم آی پی ها گروه دارن پس آدرس آی پی همیشه به دو قسمت آدرس گروه (net id) و آدرس سیستم (host id) تقسیم میشود مگر اینکه در شرایط خاص آدرس گروه نداشته باشه و همش آدرس سیستم یا host id باشه .

خب این گروه رو چی مشخص میکنه؟ چیزی به نام subnet mask

چیزی که خیلی خیلی مهمه به خاطر داشته باشید اینه که آدرس آی پی بدون subnet mask اصلا معنی نمیده چون گروهش مشخص نیست.

سابنت چطوری گروه رو مشخص میکنه؟ سابنت ساختارش شبیه آی پی هستش یعنی ۳۲ بیت و ۴ قسمت ۸ بیتی. حالا از چپ ترین بیتش هر بیتی که توی سابنت ۱ باشه همون بیت توی آی پی میشه net id و این آدرس شبکه تا اولین ۰ سابنت ادامه پیدا میکنه. به مثال زیر توجه کنید :

زیر هستش :

و ما می‌خواهیم تا این بیت مشخص شده ادرس شبکه باشه (آدرس شبکه همون ادرس گروه یا net id) :

[illegible]

یس باید همراه آی پی یک سابنت به شکل زیر بیاید:

[illegible][illegible]

یا اگر بخواهیم تا بیت مشخص شده زیر آدرس شبکه باشد:

•) •) _) •) •) •) •) _) •) •) •) •) "") •) •) •) •) _) •) •) ""

))))))

خب پس فهمیدیم net id آدرس شبکه است و host id آدرس سیستم. حالا به چه دردی میخورن؟

اول اینکه شرط اول هر ارتباط لایه^۲ی (یعنی ارتباطی که از طریق سویچ بدون وجود روتر سر راه

برقرار همیشه) هم رنج بودن آدرس سیستم ها هستند و هم رنج بودن یعنی net id یکسان باشد با host id

متفاوت

برای مثال :

192.16A.1.1

२५५ २५५ २५५ .

பா

192 168 1 2

२५५ २५५ २५५ .

هم رنجه چون net id هر دو آدرس بنا بر سابنت ماسک همراهشون ۱۹۲.۱۶۸.۱ هستش پس همرنجن
یا مثلا

۱۹۲.۱۶۸.۱.۱

۲۵۵.۲۵۵.۰.۰

با

۱۹۲.۱۶۷.۱.۲

۲۵۵.۲۵۵.۰.۰

هم رنجه نیست چون net id آدرس اول ۱۹۲.۱۶۸ ولی net id آدرس دوم ۱۹۲.۱۶۷ و چون یکسان نیستن
یعنی این آدرس ها همرنجه نیستن

چیزی که باید بهش توجه کنید اینه که اولاً گاهی سابنت ماسک رو به شکل prefix نشون میدن یعنی مثلا به
جای :

۲۵۵.۲۵۵.۲۵۵.۰

مینویسن /۲۴

یعنی به جای اینکه زیر آدرس اون عبارت دسیمال رو بنویسن در ادامه آی پی یک / قرار میدن و تعداد ۱
هایی که اون عبارت دسیمال رو ساختن رو مینویسن. برای فهم بهترش مثال های زیر رو ببینید

۲۵۵.۲۵۵.۲۵۵.۰

یک عبارت دسیمال هست که در فرم باینری به شکل زیر میاد:

۱۱۱۱۱۱۱۱.۱۱۱۱۱۱۱۱.۱۱۱۱۱۱۱۱.۰۰۰۰۰۰۰۰

توی عبارت بالا ۲۴ تا ۱ عبارت ۲۵۵.۲۵۵.۲۵۵.۰ رو ساختن پس توی شکل prefix میشه /۲۴

۲۵۵.۲۵۵.۰.۰

در فرم باینری میشه

۱۱۱۱۱۱۱۱.۱۱۱۱۱۱۱۱.۰۰۰۰۰۰۰۰.۰۰۰۰۰۰۰۰

پس توی شکل prefix میشه /۱۶

۲۵۵.۲۵۵.۱۲۸.۰

توی فرم باینری میشه

۱۱۱۱۱۱۱۱.۱۱۱۱۱۱۱۱.۱۰۰۰۰۰۰۰.۰۰۰۰۰۰۰۰

پس توی شکل prefix همیشه ۱۷/

مثال بالا نشون میده که همیشه مقدار هر اوکند ۲۵۵ نیست و اعداد دیگم میتونه باشه این به این علته که ما نباید آی پی رو به عنوان یک آدرس ۴ اوکندی ببینیم چون خود کامپیوتر با اوکند ها کاری نداره و اون یک بلوک ۳۲ بیتی رو میبینه و host id و net id بر اساس بیت ها جدا میشن نه اوکند ها یعنی چی ؟ به زبون ساده ای کامپیوتر وقتی میخواد آی پی رو چک کنه یک جعبه میبینه که ۳۲ تا چراغ کنار هم قرار دارن و از سمت چپ به تعداد سابنت ماسک می‌شماره میاد جلو و بعد اون رو host id در نظر میگیره. پس سابنت ماسک ممکنه همیشه اعداد رند ۸ ، ۱۶ و ۲۴ که به ترتیب میشن ۲۵۵.۰.۰.۰ ، ۲۵۵.۲۵۵.۰.۰ و ۲۵۵.۲۵۵.۲۵۵.۰ نباشه و مقادیر دیگه ای بیاد که بعدا مفصل راجع بهش حرف میزنیم.

آی پی ها به ۲ دسته عمومی و خصوصی (Public , private) تقسیم میشن .
آدرس های عمومی همون آدرس هایی هستن که iana میفروشه و فقط از هرکدوم یدونه موجوده. ارتباطات اینترنتی نیاز به آدرس های عمومی داره چون باید بی‌همتا باشه که بتونه توی اینترنت شمارو پیدا کنه. از طرف دیگه آدرس های خصوصی رنج های خاصی از آی پی هستن که برای مصارف داخلی مشخص شدن و مثلا توی خونه، شرکت یا کارخونه سیستم هارو با این آدرس ها به هم متصل میکنیم.
رنج آدرس های خصوصی رو پایین میگم:

از ۱۰.۰.۰.۰ تا ۱۰.۲۵۵.۲۵۵.۲۵۵

از ۱۷۲.۱۶.۰.۰ تا ۱۷۲.۳۱.۲۵۵.۲۵۵

از ۱۹۲.۱۶۸.۰.۰ تا ۱۹۲.۱۶۸.۲۵۵.۲۵۵

این آدرس ها برای ارتباطات داخلی استفاده میشن و توی دنیا بسیار تکرار شدن برای مثال آدرس ۱۹۲.۱۶۸.۱.۱ تقریبا توی تمام خونه ها روی مودم ها ست شده پس ما با این آدرس نمیتونیم بریم توی اینترنت چون بسته بازگشت ما هیچوقت به دست ما نمیرسه چون این آدرس توی همه جای دنیا هست و نیاز به آدرس عمومی داریم که بی‌همتا باشه.

دسته بندی دیگری که توی آی پی آدرس ها داریم کلاس بندی آی پی ها هستن که یک قرارداد عمومی برای مشخص کردن host id هست و امروزه دیگه مجبور به استفاده از اون نیستیم .
کلاس های آی پی به شکل زیر هستن :

از ۱.۰.۰.۰ تا ۱۲۶.۲۵۵.۲۵۵.۲۵۵ کلاس a هست که سابنتش ۲۵۵.۰.۰.۰

از ۱۲۸.۰.۰.۰ تا ۱۹۱.۲۵۵.۲۵۵.۲۵۵ کلاس b هست که سابنتش ۲۵۵.۲۵۵.۰.۰

از ۱۹۲.۰.۰.۰ تا ۲۲۳.۲۵۵.۲۵۵.۲۵۵ کلاس c هست که سابنتش ۲۵۵.۲۵۵.۲۵۵.۰

از ۲۲۴.۰.۰.۰ تا ۲۳۹.۲۵۵.۲۵۵.۲۵۵ کلاس d هستش که سابنت نداره و برای مالتی کست استفاده میشه

از ۲۴۰.۰.۰.۰ تا ۲۵۴.۲۵۵.۲۵۵.۲۵۵ کلاس e هست که سابنت نداره و برای اهداف آزمایشی مورد استفاده قرار میگیره

توجه کنید که از ۱۲۷.۰.۰.۰ تا ۱۲۷.۲۵۵.۲۵۵.۲۵۵ قابل استفاده در شبکه نیست و برای اشاره به خود یا loopback استفاده میشود.

خب حالا چیزی که مهمه بدونیم اینه که ارتباط لایه ۳ با آی پی انجام میشه پس روتر ها، کامپیوتر ها، گوشی ها و کلا هر چیزی که ارتباط لایه ۳ ای میگیره اولین چیزی که نیاز داره حتما حتما آی پی هستش و بدون آی پی کار نمیکنه .

خب بالاتر گفتیم شرط ارتباط لایه ۲ ای هم رنج بودن یعنی آدرس مبدا و مقصد یک host id یکسان داشته باشن حالا چرا؟ ارتباط لایه ۲ ای از طریق مک آدرس ها انجام میشه اما ما مک آدرس هارو مثل آی پی آدرس ها نداریم پس کامپیوتر ها باید خودشون مک آدرس هارو پیدا کنن . فرض کنید من و شما ۲ تا سیستم داریم که از طریق سوئیچ به هم متصل هستن و من میخوام برای شما بسته بفرستم. من از شما میپرسم آدرس چیه من میخوام برات یه عکس مثلا بفرستم. تو میگی ۱۹۲.۱۶۸.۱.۱ . خب حالا من میرم این آدرسی که تو دادی رو میزنم توی یک برنامه ای که مثلا عکس میفرسته. کامپیوتر تو آدرس آی پی رو میگیره و توی شبکه داد میزنه آی ایها الناس مک آدرس این آی پی آدرسی که من دادم چیه؟ اینجاست که هم جالب میشه هم پیچیده.

اگر ارتباط بین تو و مخاطبت لایه ۲ ای باشه یعنی مثلا از طریق یه سوئیچ به هم وصل باشین اون جوابتو میده و میگه من کسی هستم که تو کار داری اینم مک آدرسم بزن تو بسته ات و بسته رو برای من بفرست. توی قسمت یک فهمیدیم که بعد از اینجا چه اتفاقی میوفته و بسته ها چطوری مسیریابی میشن. حالا اگر با مخاطبت لایه ۲ ای وصل نباشی مجبوری این ارتباط رو از طریق کسی بگیری که به اون لایه ۲ ای وصله.

خب چجوری اینکارو بکنیم؟ مثلا من میخوام برم گوگل. من لایه ۲ ای به مودم وصلم تو خونم گوگل هم لایه ۲ ای به مودمش وصله تو شرکتش من چجوری به مودم اون وصل شم؟ تو به مودم خودت وصل میشی بسته ات رو لایه ۲ ای تحویل مودم خودت میگی ولی میگی من با فلانی کار دارم. چجوری اینکارو میکنی؟ اونجایی که داد میزنی مک آدرس این آی پی چیه؟ مودم جوابتو میده و مک خودش میده. میگه تو بسته ات

رو با مک آدرس لایه ۲ای تحویل من بده من بسته‌ات میفرستم برای اونی که داری لایه ۳ ای با آی‌پی صداش میزنی.

خب حالا که بسته من رسید دست مودم. مودم من با مودم گوگل یه ارتباط لایه ۳ای برقرار میکنه و بسته رو تحویل مودم گوگل میده. مودم گوگل هم لایه ۲ای به سرور وصله و بسته رو تحویل سرور میده.
(همه ی حرف های بالا مثالی و در عمل قضیه پیچیده تره)

همه حرف های بالا رو زدم که بگم توی ارتباط لایه ۲ای هم رنج بودن مهمه چون باید جواب اون سوال مک آدرس این آی پی چیه داده بشه. این سوال رو یک پروتکلی میپرسه به نام arp که کارش اینه که وقتی ما یک آی پی داریم ولی مک مرتبط با اون رو نداریم ؛ یک بسته broadcast میفرسته توی شبکه و از همه اعضا شبکه میپرسه که اگر شما صاحب آی پی آدرس داخل من هستین به این بسته پاسخ بدین تا من مک آدرس شماره پیدا کنم .

خب ما وقتی داشتیم بسته میفرستادیم ۴ تا خونه رو توی یه جدولی پر کردیم

۱. مک آدرس فرستنده : که میشه خودمون و خود کامپیوتر از روی کارت شبکه بر میداره و میزازه روی بسته.

۲. مک آدرس گیرنده : که نداریم پس میگیریم همه؛ برای ارسال بسته به همه از مک آدرس ۱۲ تا f استفاده میکنیم که تمام بیت هایش ۱ هستش و کل شبکه لایه ۲ای دریافتش میکنن.

۳. آی پی آدرس فرستنده : که آی پی آدرس خودمونه و خود کامپیوتر از روی کارت شبکه بر میداره.

۴. آی پی آدرس مقصد : که همان آیرسی است که مخاطبمان به ما میده و سیستم اون از روی این آدرس میفهمه که مخاطب ما اونه و جواب این آرپ رو میده.

خب حالا وقتی کسی که به آرپ پاسخ میده جواب مارو میده این جدول اینطوری پر میشه :

۱. مک آدرس مبدا : که میشه مک آدرس خودش و از روی کارت شبکه بر میداره (این همون مک آدرسیه که ما نداشتیم و بخاطرش آرپ ارسال کردیم و بجاش ۱۲ تا f گذاشتیم)

۲. مک آدرس مقصد : مک آدرس ما که از روی بسته آرپ ارسالی ما برداشته است.

۳. آی پی آدرس مبدا : آی پی آدرس خودش که خودکار از روی کارت شبکه‌اش برداشته شده.

۴. آی پی آدرس مقصد : که میشه آی پی آدرس ما که از بسته ارسالی آرپ برداشته.

به این پروسه میگن آرپ و تمام ارتباط لایه ۲ای بر پایه آرپ برقراره و اگر ارتباط لایه ۲ای نباشه هیچ ارتباطی نیست.

خب حالا برای اینکه اعضا شبکه جواب آرپ مارو بدن ما باید با اون ها هم رنج باشیم وگرنه بسته های broadcast مارو دریافت نمیکنن و آرپ مارو نمیگیرن. سیستم ها یا بسته هایی که به آدرس خودشون ارسال میشه رو دریافت میکنن یا بسته هایی که به آدرس broadcast شبکه خودشون ارسال شده. آدرس broadcast هر شبکه میشه net id اون شبکه به علاوه ۱ بجای تمام بیت های host id . پس دلیل اینکه شرط ارتباط لایه ۲ ای همرنج بودن اینه که آرپ broadcast عمل میکنه و شرط دریافت بسته های broadcast هم رنج بودن.

خب آدرس بعدی port number هست که آدرسینگ لایه ۴ هستش. بسته ای که ارسال میشه از طریق مک آدرسش توی سویچ مسیریابی میشه و با آی پی آدرسش میرسه به دست کامپیوتر ؛ حالا هر کامپیوتر ۱ آی پی آدرس داره (البته ۱ آی پی به ازای هر کارت شبکه داره اما توی این مثال ما سیستم های عمومی که از ۱ کارت شبکه استفاده میکنن رو در نظر گرفتیم.) و ۱۰۰۰ تا کار انجام میده. خب پس کلی بسته متفاوت داره با آی پی آدرس سیستم ما به دست ما میرسه که حالا باید توی کامپیوتر مسیریابی بشن به برنامه مورد نظرشون.

فرض کنید که شما در حال صحبت تلفنی با اسکایپ بر روی سیستمتان هستید؛ این ارتباط یکسری بسته مخصوص به خودشو ارسال و دریافت میکنه. یک کروم هم باز کردین و ۵ تا تب فعال دارین؛ کروم شما ۵ تا ارتباط مجزا از هم داره که باید بسته هاش جدا از بسته های اسکایپ توی سیستم شما مسیریابی بشن. پس آدرسینگ لایه ۴ ای داخل سیستم ما بسته هارو به برنامه ها و حتی در صورت نیاز در داخل یک برنامه به تب های مختلفش تحویل میده.

پورت ها شکل و ساختار متفاوتی با مک و آی پی دارن. هر آی پی ۵۶۵۳۵ تا پورت داره که برای هر ارتباط یک port number متفاوت استفاده میشه.

شماره پورت به همراه آدرس ها بعد از ۲ نقطه می آید :

برای مثال برای ای پی آدرس ها :

۱۹۲.۱۶۸.۱.۱:۱۲۳۴

۱۲۷.۰.۰.۱:۴۴۳

۸.۸.۸.۸:۵۰۶۰

یا برای url ها :

Google.com:80

Yahoo.com:443

(آدرس های url در اصل فقط واسطه هایی هستند که ip آدرس هارو صدا میزنند و چون به خاطر سپردن آی پی آدرس ها سخت از url به جای اون ها استفاده میکنیم اما در اصل ارتباط از طریق ip انجام میشود.) از پورت ۰ تا ۶۵۵۳۵ برای اهداف خاص رزرو شدن و برای ارتباطات مشخصی استفاده میشن.

(ما هم میتونیم از این پورت ها برای کار های دیگه استفاده کنیم اما ممکنه اگر سرویسی داره از well known port استفاده میکنه باهش تداخل پیدا کنه)

توصیه میکنم well known ports رو سرچ کنید و پورت های معروف رو مطالعه کنید

برای هر ارتباطی در شبکه باید port number بدیم و بدون شماره پورت ما در اصل کاری که با اون مخاطب داریم رو مشخص نکردیم.

بزارید یک مثال بی ربط بزنم :

فرض کنید شما از من یک لیوان آب میخواید و فقط داد میزنید : شاهین!!!!

آی پی آدرس مثل اسم میمونه. شما وقتی آی پی آدرس رو صدا میزنید در اصل فقط دارید توجه اون سیستم رو به خودتون جلب میکنید اما با شماره پورت کارتونو ازش میخواید.

شما باید داد بزنید : شاهین ، به من آب بده !

شاهین = آی پی آدرس

به من آب بده = شماره پورت

بنا بر این حرف ها ما هرکاری توی اینترنت انجام میدیم باید شماره پورت بدیم؛ برای مثال ارتباط با یک وب سایت از طریق شماره پورت های ۸۰ و ۴۴۳ انجام میشه و ما وقتی میخوایم مثلا گوگل رو باز کنیم باید بزنیم :

Google.com:80

ولی ما فقط میزنیم google.com و باز میشه چجوری:

خب اینجا well known port ها به ما کمک میکنن تا سرویس های بین المللی ارائه بدیم. یعنی چی ؟

یعنی اینکه مثلا بجای اینکه بگیم اگر میخوای سایت منو باز کنی بزن Shahin.com:80

میگم بزنی shahin.com و مرورگرم خودکار :80 رو اضافه میکنه ؛ چرا؟ چون well known port برای ارائه وب سایت ۸۰ هستش و اینو همه میدونن و مرورگر ها خودشون اضافه میکنن. حالا ما نمیتونیم یک وب سایت رو روی ۸۰ ارائه ندیم؟ چرا میشه ولی اونجوری باید به همه بگین پورتنون چنده چون مرورگرها پیشفرض پورت ها ۸۰ و ۴۴۳ رو استفاده میکنن. پس فقط میتونید توی سرویس های محلی سرویس هارو روی پورت های دلخواه ارائه بدین.

خب حالا اینکه ما میتونیم همزمان کارهای اینترنتی متفاوتی رو روی کامپیوترمون انجام بدیم بخاطر پورت هاست. بالاتر گفتیم که ما موقع ارسال بسته ها یک جدولی رو پر میکنیم که ۴ تا خونه داره. این جدول ۲ تا خونه دیگه هم داره به نام های :

۱. شماره پورت فرستنده : که عموماً یک عدد رندوم توی یک رنجی که همون برنامه مشخص میکنه و

برای اینکه که توی بسته پاسخ بسته درست به دست صاحبش برسه

۲. شماره پورت گیرنده: که کاری که ما با مخاطبمون داریم رو با اون مشخص میکنیم.

یه قابلیت دیگه ای که port number برای ما فراهم میکنه nat یا network address translate .

این قابلیت به ما اجازه میده با استفاده از پورت، یک آدرس رو به آدرس دیگه ای ترجمه کنیم؛ به چه کاری میاد؟ گاهی ما یک آدرس داریم ولی نیاز داریم با اون یک آدرس به کلی ادم اینترنت بدیم؛ خب ما میدونیم هر کس برای ایجاد ارتباط اینترنتی نیاز به یک آدرس عمومی داره اما خب ما ۱ مودم داریم و ۱ آدرس عمومی؛ چاره چیه؟ ما باید همرو با آدرس های خصوصی به هم متصل کنیم و موقعی که میخوان برن توی اینترنت آدرس های خصوصیشون رو به آدرس عمومی مودم ترجمه کنیم که اینکار از طریق Nat انجام میشه.

نت ها به دو دسته source nat و destination nat تقسیم میشوند که همونطور که از اسمشون پیداست

source nat قسمت source ip رو ترجمه میکنه و destination nat قسمت destination ip رو.

اگر بخوایم خیلی کلی و یکم غلط تعریفش کنیم میتونیم بگیم ما موقعی که میخوایم بریم توی اینترنت مشکل source ip داریم چون آدرس داخلیه. پس باید از source nat استفاده کنیم و بالعکس وقتی میخوایم از اینترنت به سمت شبکه داخلی بیایم چون ما فقط میتونیم آدرس عمومی روتر رو صدا بزنیم اما با یک آدرس داخلی، داخل شبکه کار داریم پس مشکل از source ip ماست که باید از یک آدرس عمومی به آدرس خصوصی تغییر کنه از destination nat استفاده میکنیم .

حالا این نت چجوری کار میکنه ؟ اینو با یه مثال توضیح میدم اما میتونه مطلب دشواری باشه و بهتره چند بار بخونید تا بهتر بفهمید:

فرض کنید شما از داخل خونه با گوشی موبایلتون میخواید برید گوگل. خب اولاً ما میدونیم که گوشی ما تو خونه به مودم وصله؛ خود مودم یک آدرس عمومی داره مثلاً ۲.۳.۴.۵ که روی پورته ست شده که به سمت اینترنت میره مثلاً توی مودم های adsl روی اون پورت تلفنه، خب ارتباط بین گوشی ما و مودم خونه با آدرس خصوصی انجام میشه، مثلاً مودم آدرس ۱۹۲.۱۶۸.۱.۱ رو داره و گوشی ما هم ۱۹۲.۱۶۸.۱.۲ . خب وقتی من از گوشیم به گوگل که مثلاً آدرسش ۱.۲.۳.۴ هست بسته میفرستم اون جدول پر میشه :

۱. مک فرستنده : مک خودم

۲. مک گیرنده : مک مودم (قبلاً گفتیم چرا مک مودم)

۳. آدرس آی پی فرستنده : ۱۹۲.۱۶۸.۱.۲

۴. آدرس آی پی گیرنده : ۱.۲.۳.۴

۵. پورت فرستنده : ۲۴۵۳۱ (عموماً src port یک عدد رندوم توی یک رنج خاص مربوط به هر برنامه است)

۶. پورت گیرنده : ۸۰ (چون مثلاً داریم میریم وب سرویش رو باز کنیم)

خب حالا بسته ما میرسه به دست مودم و مودم مبخواد ارسالش کنه ؛ آدرس فرستنده بسته ما یک آدرس خصوصی و توی جواب بسته جای آدرس فرستنده و گیرنده با هم عوض میشه و خب میدونیم که بسته ما با آدرس خصوصی برنمیگرده سمت ما. پس ما میریم روی مودم میگیریم مثلاً هر وقت یک بسته خواست از اون پورت تلفن که به سمت اینترنت بره بیرون روش عمل `source nat` انجام بده و `source ip address` رو با آدرس ۲.۳.۴.۵ که آدرس عمومی مودمونه عوض کن.

خب حالا مودم میگه اوکی من یه جدولی دارم به نام جدول نت (`nat table`) آدرس بسته تو عوض میکنم ولی میرم توی جدول مینویسم بسته که برگشت آدرس خودشو میزارم سر جاش؛ آدرس و چجوری عوض میکنم؟ خیلی سادست بسته تو آدرس های مبدا اینجوریه :

۱۹۲.۱۶۸.۱.۲:۲۴۵۳۱

من تبدیلتش میکنم به آدرس زیر :

۲.۳.۴.۵:۱۴۳۲۵

بعد میرم توی جدول نت عبارت زیر رو اضافه میکنم :

۱۹۲.۱۶۸.۱.۲:۲۴۵۳۱ <-- ۲.۳.۴.۵:۱۴۳۲۵

بعد بسته رو ارسال میکنم.

هر موقع یه بسته به دستم رسید که مقصدش ۲.۳.۴.۵:۱۴۳۲۵ بود من میگیرمش عوضش میکنم به

۱۹۲.۱۶۸.۱.۲:۲۴۵۳۱ و بعد تحویل صاحبش میدم.

میدونم پیچیده شد اما اگر چند بار بخونیش و برای خودت روی کاغذ بنویسیش میفهمی.

خب حالا یه مثال از `destination nat` ببینیم :

فرض کنید ما ۵ تا سرور داریم که سرویس وب ارائه میدن و یک مودم که اینترنت داره و میخوایم این ۵ تا سرویس رو از طریق اینترنت ببینیم ؛ ما میدونیم که وب سرویس ها روی پورت ۸۰ و ۴۴۳ ارائه میشن پس ما برای اینکه این سرویس ها رو باز کنیم باید آدرسشونو به همراه پورت مثلاً ۸۰ بزنیم. حالا ما یه آدرس عمومی روی روتر داریم و ۳ تا سرویس چکار کنیم؟ از `destination nat` استفاده میکنیم.

فرض کنید آدرس عمومی مودم ۱.۲.۳.۴ و آدرس خصوصی سرور ها ۱۹۲.۱۶۸.۱.۱ ، ۱۹۲.۱۶۸.۱.۲ و ۱۹۲.۱۶۸.۱.۳ و همشون روی پورت ۸۰ سرویس میدن .

ما روی مودم ۵ تا `destination nat` به شکل زیر تعریف میکنیم :

۱۹۲.۱۶۸.۱.۱:۸۰ <-- ۱.۲.۳.۴:۱۲۳۴

یعنی هرکس آدرس عمومی مارو به همراه پورت ۱۲۳۴ صدا زد منظورش ۱۹۲.۱۶۸.۱.۱:۸۰

۱۹۲.۱۶۸.۱.۲:۸۰ <-- ۱.۲.۳.۴:۲۳۴۵

یعنی هرکس آدرس عمومی مارو به همراه پورت ۲۳۴۵ صدا زد منظورش ۱۹۲.۱۶۸.۱.۲:۸۰

۱۹۲.۱۶۸.۱.۳:۸۰ <-- ۱.۲.۳.۴:۳۴۵۶

یعنی هرکس آدرس عمومی مارو به همراه پورت ۳۴۵۶ صدا زد منظورش ۱۹۲.۱۶۸.۱.۳:۸۰
و ما به این شکل میتونیم ۱ آدرس عمومی رو به ۴ آدرس خصوصی ترجمه کنیم .

خب همینجا بحث پورت رو میبندم و میرم سراغ url ها

آدرس های url آدرس هایی هستند که برای کمک به آی پی آدرس ها استفاده میشن یعنی چی؟
خب ما میدونیم که هر سروری یک آدرس آی پی داره و ما برای ارتباط با اون سرور باید آی پی آدرسشو
صدا بزنیم حالا فرض کنید برای هر سایتی که بهش مراجعه میکنید مجبور بودین ۱ آی پی حفظ کنید ،
اینکار بسیار سخته به زمانی فکر کنید که موبایل ها این شکلی در دسترس نبودن و ما باید شماره تلفن هارو
حفظ میکردیم؛ پس ما نمیتونیم برای facebook.com ، yahoo.com ، google.com و ... آی پی
حفظ کنیم اما یک آدرسی که از اسمشون تشکیل شده رو میشه حفظ کرد. اینجاست که url ها به کمک آی پی
ها میان و ما وقتی با یه سایتی کار داریم url اون سایت رو صدا میزنیم و یک سرویسی به نام dns یا
domain name service این url رو به یک آی پی ترجمه میکنه و ما اون سایت رو باز میکنیم.