آران پارت آسیا
ماشین آلات راهسازی، معدنی و ساختمانی

تاریخ: / /

۹۸۲۳۰۳۳ - راسخانی

۱- الف) پروتکل UDP برخلاف TCP که می‌تواند تضمین کند که بسته‌ها حتما ارسال شوند یا نشوند، UDP unreliable است که تلاشش را می‌کند تا بسته‌ها درست و بدون خطا و به ترتیب به دست گیرنده برسند اما تضمین نمی‌دهد. به این دلیل که UDP connectionless هست هیچ سیگنال کنترلی در ابتدای کار بین فرستنده و گیرنده رد و بدل نمی‌شود و بسته به صورت کاملا مستقل رفتار می‌شود. حال اگر نیاز به سرویسی داشته باشیم که بتواند کنترل بسته‌ها را شخصی درست کند باید خودمان در لایه یا لایه‌ای بالاتر (لایه اپلیکیشن) پیاده‌سازی کنیم تا انتقال داده مطمئن داشته باشیم.

الف - ب) ممکن است سیگنالی که ارسال می‌شود بعضی از بیت‌هایش flipped شوند یعنی صفرها به یک تبدیل شده و یک‌ها صفر شوند پس checksum می‌تواند ما را خطاهای از این دست را تشخیص دهد.

برای اینکه از مشکل ارسال بسته‌های تکراری جلوگیری کنیم می‌توانیم از sequence number. سمت فرستنده استفاده کنیم. یعنی بسته‌هایی که ارسال می‌کنیم شماره‌گذاری کنیم. در این صورت سمت گیرنده شماره بسته‌های دریافتی را چک کرده و اگر بسته تکراری است به جای اینکه به لایه اپلیکیشن تحویل دهد آن را دور بریزد. برای شماره‌گذاری بسته‌ها (sequence number) فقط یک بیت (همان 0 و 1) کفایت می‌کند چون فرض می‌کنیم stop and wait تنها یک بسته ارسال می‌شود اگر NACK و یا یک بسته خراب شده دریافت شد که مشخص نیست ACK است یا NACK) دوباره همان بسته را ارسال می‌کند تا اینکه NACK دریافت شود

تاریخ : / /

آران پارت آسیا
ماشین آلات راهسازی، معدنی و ساختمانی

که بسته جدید ارسال می‌کند. این به این معنی است که بسته‌ی قبلی ارسال می‌شود تا یک بسته‌ی جدید ارسال می‌شود. همچنین sequence number می‌تواند از گم شدن بسته‌ها جلوگیری کند.

۱-ج) برای مواجه با گم شدن بسته‌ها از timer استفاده می‌شود. پس طول بازه timeout هم است. این زمان باید از RTT بیشتر باشد. اگر مقدار اولیه timer را خیلی کوچک انتخاب کنیم باعث می‌شود تا بسته‌ها زودتر از موعد مورد نیاز timeout شده و این موضوع می‌تواند منجر به بازارسال‌های بی‌جهت شود. اما اگر این مقدار خیلی بزرگ باشد باعث می‌شود اگر گم شدنی در شبکه رخ دهد عکس‌العمل ما برای ارسال مجدد کند بوده و نمی‌توانیم از منابعی مانند ظرفیت کانال به درستی استفاده کنیم.

تاریخ: / /

۱- ۲)

A = 7BEØ B = 8653

```
        1 1 1 1 1 1 1
A =   0 1 1 1 / 1 0 1 1 / 1 1 1 0 / 0 0 0 0
B =+  1 0 0 0 / 0 1 1 0 / 0 1 0 1 / 0 0 1 1
      ---------------------------------------
Carry (1) 0 0 0 0 / 0 0 1 0 / 0 0 1 1 / 0 0 1 1
          |+                              1 1 |
          ---------------------------------------
Sum   =   0 0 0 0 / 0 0 1 0 / 0 0 1 1 / 0 1 0 0
```

مکمل یک می‌گیریم.

Checksum = 1111 1101 0001 0100 = FDCB

F D C B

۱-ه) اگر پیام Ack دریافت نشود می تواند علت تأخیر در ارسال آن

یا نه دریافت شدن آن باشد.

که اگر به علت دریافت نشدن آن باشد که timer آن را مدیریت کرده و با timeout

مجدداً بسته ارسال شده و نیازی به sequence number نیست (چون بسته تکراری

یا duplicate شده نفرستادیم راست ه).

اما اگر ack به علت تأخیر در ارسال مجدداً بسته ای ارسال میشود (بسته

تکراری) این مشکل با seq number حل و مدیریت میشود.

A = Source port = 6507 (i=1

dest port = 6582

B = source Port = 6582 dest Port = 6507

C = source Port = 5301 dest port = 6582

D = source Port = 5333 dest Port = 6582

۲ـ

الف) پورت مبدأ = ۳۹۷c پورت مقصد: 01bb

ب) منتظر بایت (b53d 5600) هستیم یعنی ما قبل از آن را ACK کردیم

ردیف ۷: (0101 0000 0001 0001 0000 0010 0000 0011)

A FIN rcv window

flag A = 1 یعنی نشان دهنده معتبر بودن است.

ج) flag FIN = 1 یعنی این بسته سیگنال نشتی داشته و مربوط به قطع دو مبدأ ارتباط است.

د) seq number، b6 c1 40 86 است. این عدد شماره اولین بایت داده را مشخص می‌کند

۳- الف) پورت مبدأ = ۵۴۴۵ ، seq number = ۴۴۷

پورت مقصد: ۸۰

ب) پورت مبدأ = ۸۰

پورت مقصد: ۵۴۴۵ ، Ack number = ۲۰۷

ج) پورت مبدأ = ۸۰

پورت مقصد: ۵۴۴۵ ، Ack number = ۱۲۷

د) Ack number بسته اول = ۲۰۷ ، Ack number بسته دوم = ۳۲۷
= ۴۴۷ + ۸۰

Ack number بسته سوم = ۱۲۷

هـ) بسته ①: sequence number = ۱۲۷ ، ۸۰ بایت

بسته ②: seq number = ۲۰۷ ، ۴۰ بایت

بسته ③: seq number = ۴۴۷ ، ۸۰ بایت

به دلیل عدم دریافت Ack بسته اول و تمام شدن زمان این بسته باز ارسال می شود (timeout شدن)

باز ارسال بسته ①: seq number = 127 ، ۸۰ بایت

اما سمت B این بسته تکراری بوده (duplicate) شمرده شده

① Ack = Ack number = ۲۶۷

② Ack = Ack # ۲۶۷

③ Ack = Ack # ۳۲۷

④ Ack = سمت B بسته تکراری دریافت شده بود آن بسته را دور ریخته و آخرین Ack را شنیده دوباره ارسال می‌کند یعنی Ack # ۳۲۷ را مجدداً ارسال می‌کند.

-۸

اولیه Estimated RTT = ۱۲۰ ms ، اولیه Dev RTT = ۵ ms ، α = ۰/۱۲۵ ، β = ۰/۲۵

DevRTT = (1-β) × DevRTT + β × |SampleRTT - EstimatedRTT|

estimatedRTT = (1-α) × estimatedRTT + α × samRTT

TimeoutInterval = EstimatedRTT + 4 × DevRTT

RT = ۱۲۸ ms :

estimatedRTT = (1 - ۰/۱۲۵) × (۱۲۰ms) + (۰/۱۲۵ × ۱۲۸)

= ۱۰۵ + ۱۶ = ۱۲۱ ms

DevRTT = (1 - ۰/۱۲۵) × ۵ms + ۰/۱۲۵ |۱۲۸ - ۱۲۰|

= ۴/۷۵ + ۲ = ۵/۷۵ ms

RT = ۱۲۵ ms

estimatedRTT = (1 - ۰/۱۲۵) × (۱۲۱) + (۰/۱۲۵ × ۱۲۵)

= ۱۰۵/۸۷۵ + ۱۶/۸۷۵ = ۱۲۲/۷۵ ms

DevRTT = (1 - ۰/۱۲۵) × ۵/۷۵ + (۰/۱۲۵ × |۱۲۵ - ۱۲۱|) =

= ۴/۳۱۲۵ + ۳/۵ = ۷/۸۱۲۵ ms

$R_3 = 151\ ms$

$$estimatedRTT = (1 - 0.125) \times 137.75 + 0.125 \times 151 =$$

$$107.45 + 17.875 = 125.325\ ms$$

$$DevRTT = (1 - 0.125) \times 7.81 + 0.25 \times |151 - 137.75|$$

$$= 6.835 + 3.375 = 10.21\ ms$$

$R_4 = 110\ ms$

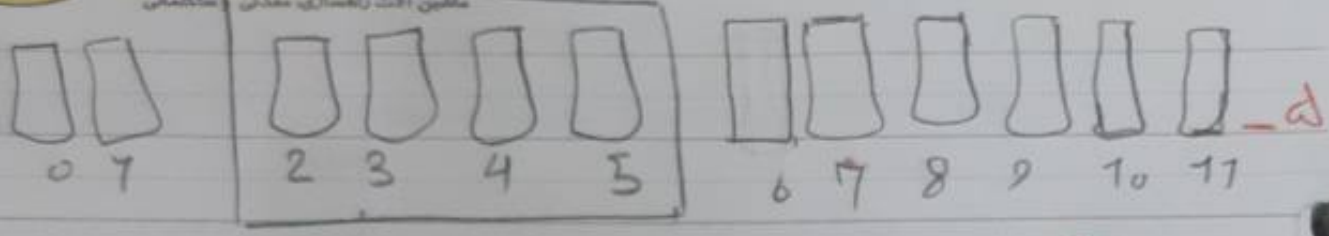
$$estimatedRTT = (1 - 0.125) \times 125.325 + 0.125 \times 110 =$$

$$109.659 + 13.75 = 123.409\ ms$$

$$DevRTT = (1 - 0.25) \times 10.21 + 0.25 \times |110 - 125.325|$$

$$= 7.658 + 3.831 = 11.489$$

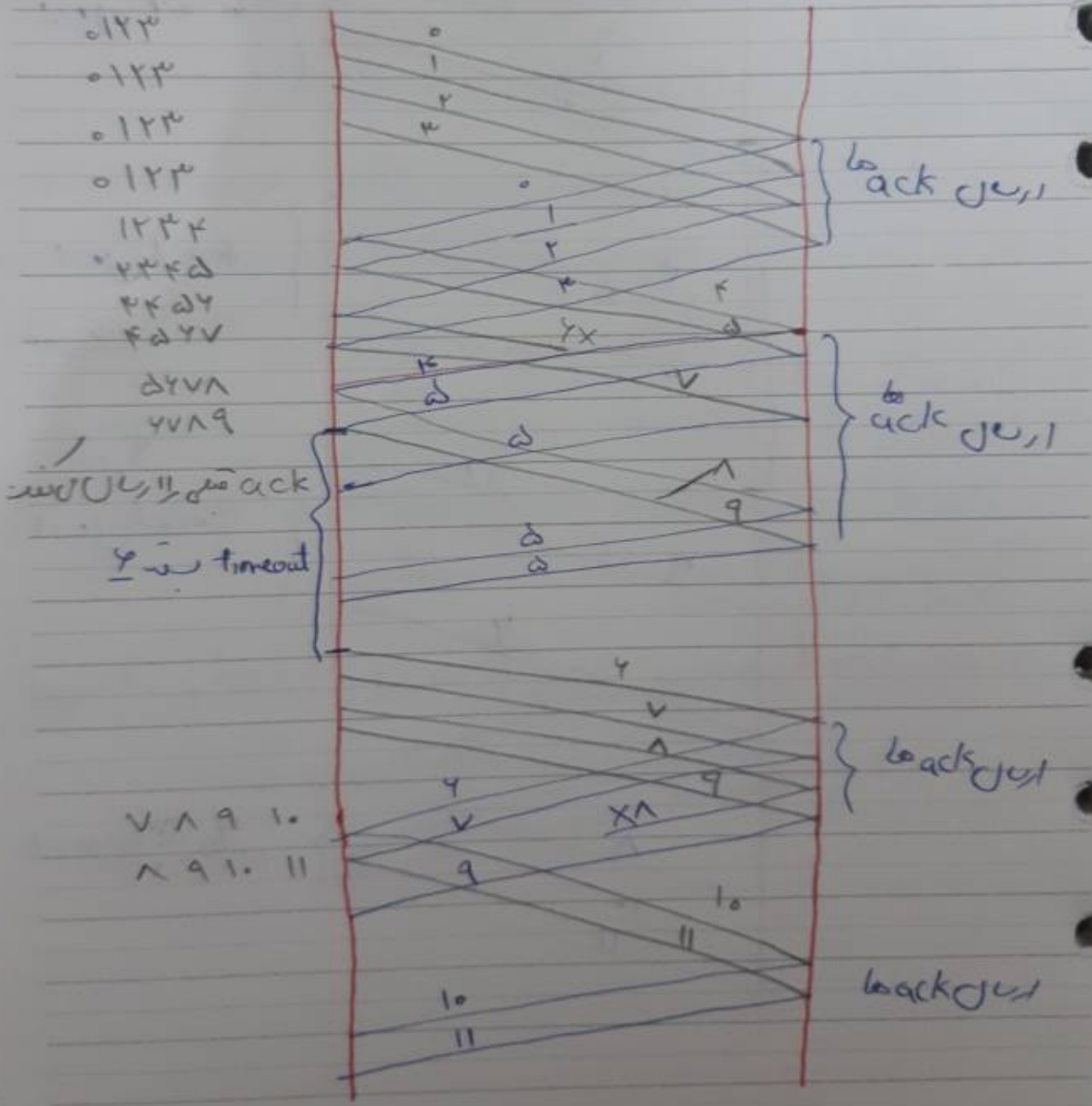
$$Time\ out\ interval = 123.409 + 4 \times 11.489 = 169.365\ ms$$



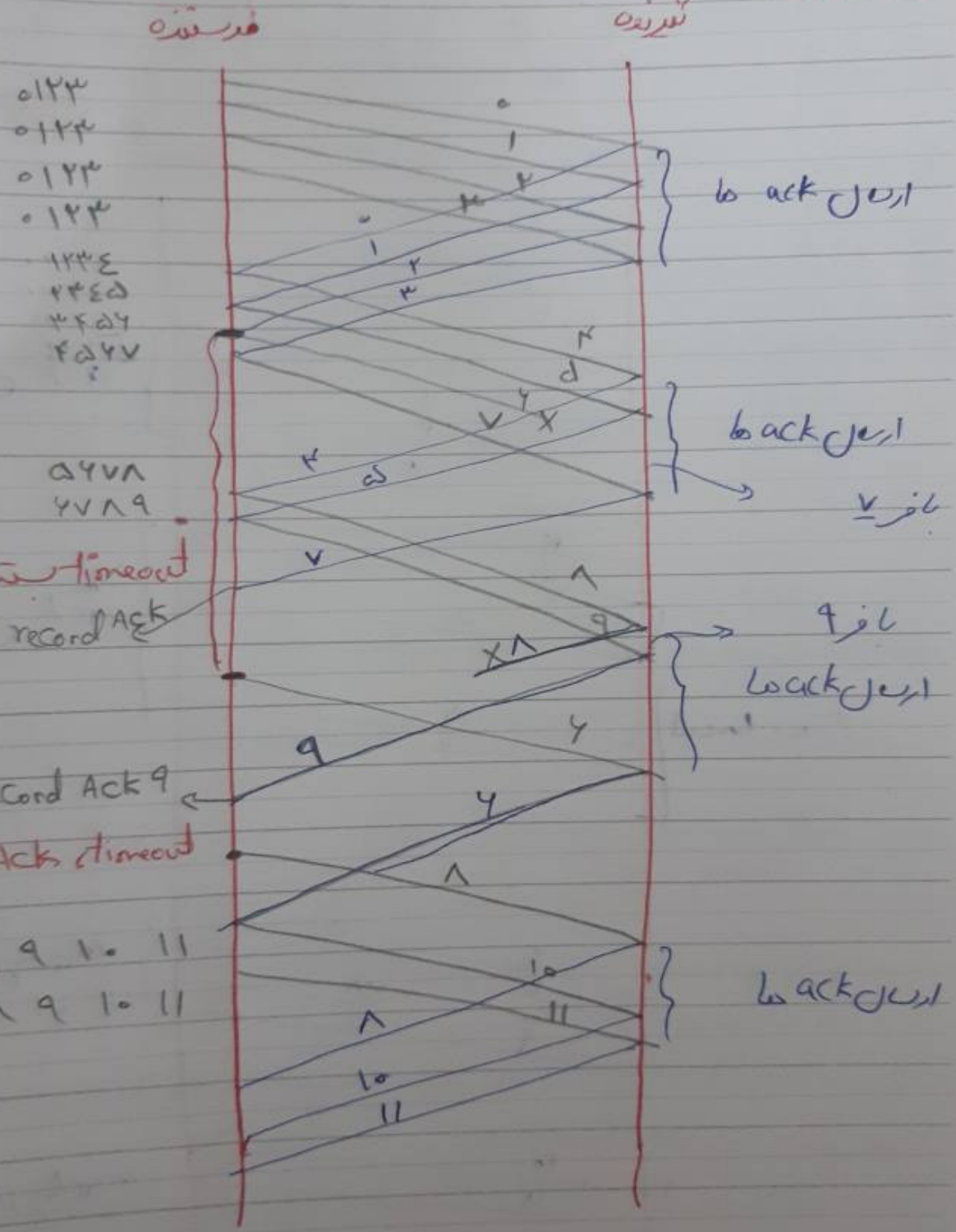
۵_

حالت GBN

فرستنده | گیرنده



حالت SR :



الف) در حالت GBN، Ack های تکراری دریافت می‌شود پس با timeout شدن دوباره پنجره ۷ و همه بسته های داخل window دوباره بازارسال می‌شوند.

در حالت SR پس از timeout شدن مربوط به بسته ۷ فقط همین بسته مجدد ارسال می‌شود چون بسته های بعدی دریافت شده هستند.

ب) در حالت GBN، بازارسالی نداریم چون Ack ۹ به فرستنده رسیده و این Ack حالت تجمعی دارد پس در واقع Ack مابقی window نیز ارسال می‌شود.
اما در حالت SR می‌دانیم که Ack ها تجمعی نیستند پس اگر این Ack گم شود پس از timeout شدن مربوط به بسته ۸ این بسته مجدد ارسال می‌شود اما گیرنده آن را دریافت خواهد داشت پس برای جلوگیری از duplicate شدن بسته ها آن را دور ریخته و Ack مربوطه را ارسال می‌کند.

ج) در حالت GBN این مقدار برابر با $2.5RTT = 1RTT + 1.5RTT$ است اما در حالت SR این مقدار برابر با $1.5RTT$ خواهد شد.

سوال ۶:

الف) NMAP یک اسکنر بسیار قدرتمند و یک نقشه یاب است که به منظور کاوش در بررسی امنیتی شبکه به کار می رود و یک ابزار open source یا به اصطلاح متن باز است که برای اسکن و جستجوی آسیب پذیری ها مورد استفاده امنیت کاران و هکرها قرار می گیرد.

برخی از امکانات NMAP عبارتند از:

اسکن یک میزبان - اسکن پی ای ها - پورت اسکن - اسکن سایبرنت ها - آنالیز پی ای ورژن ۶ - اسکن پاکت های خام - بایپس Firewall - اسکن پورت توسط پروتکل - حمل مک آدرس - whois

آموزش کار با این نرم افزار (برخی دستورات مهم):

ساده ترین دستور انجام بررسی پورت های یک پی ای است که به صورت زیر استفاده می شود.

برای مثال برچ شبکه ۱۹۲.۱۶۸.۱.۰ است که می خواهیم آن را اسکن کنیم:

nmap 192.168.1.0/2

این دستور به صورت پیش فرض ۱۰۰۰ پورت اول را اسکن می کند.

اما وقتی بخواهیم تمام پورت های یک آیپی را مورد بررسی و اسکن قرار دهیم از سوییچ PI استفاده می کنیم.

```
nmap -p1-65535 192.168.1.2
```

اسکن سریع یک سیستم:

```
nmap -F 192.168.1.1
```

اسکن سریع همراه با ارائه جزئیات و نسخه پورت ها:

```
nmap -v 192.168.1.1
```

شناسایی سیستم عامل همراه با اسکن با جزئیات:

```
nmap -v -A 192.168.1.1
```

اسکن کامل با بازه دلخواه شماره پورت ها:

```
nmap -v -p 1-100 192.168.1.1
```

در این دستور تنها پورت های بازه 1 تا 100 اسکن شده و گزارش می شوند

اسکن با ارسال بسته های تقسیم شده جهت جلوگیری از بلاک شدن در فایروال سیستم مقصد:

```
nmap -v -f
```

ارسال بسته اتک پورت خاص جهت اسکن ناشناس:

```
nmap -v 192.168.1.1 -g 80
```

در این دستور nmap تمامی بسته ها را از طریق پورت ۸۰ ارسال می کند و سیستم مقصد به تصور یک درخواست وب به آن نگاه می کند

اگر نخواهیم در صورتی که سیستم مقصد Ping را بلاک کرده باشد:

nmap -v -Pn

این دستور اقدام برای چک کردن وضعیت سیستم از پروتکل ICMP استفاده می‌کند و در صورتی که کاربر این پروتکل را بلاک کرده باشد از سوییچ Pn- استفاده کنیم.

ب) Ack scan: آپشن PA- منجر انجام یک اسکن TcpAckPing بروی میزبان مدنظر می‌شود. به این منظور از رشته زیر استفاده می‌شود:

sudo nmap -PA [پورت 2، پورت 1] [میزبان هدف]

گزینه PA- منجر می‌شود که nmap بسته‌های TcpAck را به میزبان‌های اختصاص داده شده ارسال کند. این بسته‌ها پاسخ به تلاش‌های TCP نامعمول برای دریافت و کشف میزبان‌های اختصاص داده شده می‌شود. مشابه گزینه‌های دیگر Ping که بسته ICMP پیش‌فرض ارسال می‌شود می‌توان از این گزینه به عنوان یک جایگزین استفاده کرد.

گزینه PA- به صورت پیش‌فرض پورت ۸۰ را اسکن می‌کند. برای اسکن پورت‌های اختصاص پورت‌ها را به‌تنهایی استفاده می‌شود:

sudo nmap -PA22,25,80,443 192.168.1.7

تاریخ: / /

Tcp syn ping گزینه PS- منجر انجام یک اسکن Tcp syn scan:

بررسی میزبان هدف می‌شود. رشته‌ای که به این منظور استفاده می‌شود به صورت زیر می‌باشد:

sudo nmap -PS [پورت آ و پورت 2 و غیره] [میزبان هدف]

Tcp syn پیغ یک بسته SYN به سیستم هدف ارسال می‌کند و منتظر پاسخ می‌ماند. این متد خاصیتی که دشمن برای سیستم‌های مفید است که برای بلاک کردن پیغ‌های استاندارد ICMP پیکربندی شده‌اند.

پورت پیش‌فرض ~~پیکربندی~~ در اسکن به وسیله PS- پورت ۸۰ می‌باشد ولی می‌توان پورت‌های دیگر نیز به آن اختصاص داد به صورت زیر

sudo nmap -PS 22, 25, 80, 473 192.168.1.7