

بسمه تعالی



آزمایشگاه شبکه
دانشکده برق و کامپیوتر
دانشگاه صنعتی اصفهان

بهار ۱۴۰۲
دکتر حیدرپور، دکتر فانیان

پیش گزارش آشنایی با امنیت پورت (port security) در سوئیچ

فهرست:

حملات لایه MAC

Port Security

حملات VLAN

Switch Spoofing

Double Tagging

دفاع در برابر حملات VLAN

لیست کنترل دسترسی

دستورات مربوط به ساخت و حذف یک لیست کنترل دسترسی

Wildcard Mask

مثال‌هایی از استفاده از ACL

امنیت مدیریت سوئیچ‌ها به کمک ACL

در معماری‌های شبکه، معمولا کاربران می‌توانند از طریق لایه access به شبکه متصل شده و از این طریق دستگاه‌های مختلفی به شبکه متصل می‌شوند که ممکن است امنیت شبکه و کاربران موجود در شبکه را به خطر بیاندازد. از این رو بررسی امنیت و به کارگیری تدابیر امنیتی برای جلوگیری از نقض امنیت از سوی کاربران در این لایه از شبکه اهمیت دارد.

می‌توان حملات قابل انجام در لایه ۲ را به چهار دسته‌ی کلی تقسیم کرد:

- حملات لایه فیزیکی (MAC Layer Attacks)
 - حملات در شبکه‌های محلی مجازی (VLAN Attacks)
 - حملات جعل (Spoofing Attacks)
 - حملات علیه سوئیچ‌ها (Switch Device Attacks)
- در این جلسه از آزمایشگاه، با حملات لایه فیزیکی (MAC) و حملات شبکه‌های محلی مجازی (VLAN) و راهکارهای موجود برای دفاع در برابر این حملات آشنا خواهید شد و در جلسه‌ی بعدی با دو دسته‌ی بعدی حملات آشنایی پیدا خواهید کرد.

حملات لایه MAC

حملات لایه MAC نوعی از حملات شبکه هستند که لایه MAC مدل OSI را هدف قرار می‌دهند. یکی از حملات این لایه، حمله‌ای با نام MAC Address Flooding است. در این حمله، مهاجم تعداد بسیار زیادی فریم با آدرس‌های فیزیکی مبدا نامعتبر و جعلی روی شبکه ارسال می‌کند. با این کار، جدول MAC موجود در سوئیچ (که آدرس‌های فیزیکی پشت هر پورت را ذخیره می‌کند) مسموم می‌شود و پس از مدتی، سطرهای آن با آدرس‌هایی که مهاجم تولید کرده جایگزین می‌شود. با پر شدن MAC Table با آدرس‌های غیر معتبر، سوئیچ به این دلیل که آدرس مقصد را در جدول خود ندارد، هر بسته‌ی دریافتی از کاربران معتبر را روی همه‌ی پورت‌های خروجی ارسال می‌کند. در این صورت مهاجم قادر خواهد بود کل ترافیک شبکه را دریافت کند. برای مقابله با این حمله، می‌توان مکانیزم Port Security را به کار گرفت که در ادامه معرفی می‌شود.

Port Security

Port Security برای مقابله با حمله ذکر شده در قسمت قبل، این امکان را به ما می‌دهد که حداکثر آدرس‌های قابل یادگیری روی یک پورت را محدود کنیم. همچنین راهکار دیگر این است که برای هر پورت، آدرس‌های فیزیکی مجاز به اتصال به شبکه از طریق این پورت را مشخص کنیم.

در Port Security، تعیین آدرس‌های فیزیکی قابل اتصال به پورت می‌تواند به صورت ایستا (static)، پویا (dynamic) یا چسبنده (Sticky) باشد. در حالت ایستا، این آدرس‌ها به صورت دستی مشخص می‌شود. در این حالت، آدرس‌های فیزیکی مشخص شده در running config سوئیچ قرار می‌گیرند که با ریست شدن سوئیچ، آدرس‌های وارد شده حفظ خواهند شد (توجه فرمایید برای این امر باید از دستور write memory استفاده نمایید).

در روش پویا، خود سوئیچ آدرس‌های فیزیکی روی یک پورت مشخص را در یک بازه زمانی یاد گرفته و از این پس بر اساس این آدرس‌ها مجاز بودن اتصال به شبکه را مشخص می‌کند. به عنوان مثال، برای یک پورت در نقطه اتصال (Access Point)، حداکثر آدرس‌های فیزیکی قابل یادگیری را برابر با ۱۰۰ قرار می‌دهیم و سوئیچ در یک بازه مشخص این آدرس‌ها را یاد گرفته و بعد از تمام شدن این زمان، سوئیچ آدرس جدیدی را قبول نمی‌کند. در بازه یادگیری باید مطمئن باشیم که شبکه قابل اعتماد است و مهاجم در شبکه حضور نخواهد داشت. همچنین در حالت پویا، با خاموش و روشن شدن سوئیچ، همه آدرس‌های فیزیکی یاد گرفته شده از بین می‌رود.

حالت سوم، حالت چسبنده است که مانند حالت پویا آدرس‌ها را در یک بازه مشخص یاد گرفته و مانند حالت ایستا این آدرس‌ها را در running config ذخیره می‌کند و با ریست شدن سوئیچ، آدرس‌های یاد گرفته شده حذف نمی‌شوند (توجه فرمایید برای این امر باید از دستور write memory استفاده نمایید).

Switch (config-if)# switchport port-security	دستور فعال کردن Port Security
Switch (config-if)# switchport port-security maximum [number]	مشخص کردن حداکثر آدرس‌های فیزیکی قابل یادگیری روی پورت
Switch (config-if)# switchport port-security mac-address [mac-address]	مشخص کردن آدرس‌های فیزیکی مجاز برای پورت به صورت ایستا

Switch (config-if)# switchport port-security mac-address sticky	فعال کردن حالت sticky برای آدرس‌های فیزیکی
Switch# show port-security interface [interface-id] [address]	مشاهده تنظیمات پورت سکیوریتی

برای مشخص کردن واکنش مناسب در هنگام عبور تعداد آدرس‌های فیزیکی دیده شده روی پورت از مقدار maximum number، یا دیده شدن آدرسی غیر از آدرس‌های فیزیکی مجاز در حالت ایستا یا پویا می‌توان از دستور زیر استفاده کرد.

```
switch(config-if)# switchport port-security violation [shutdown | restrict | protect]
```

با انتخاب حالت shutdown، در صورت رخ دادن یکی از دو شرایطی که ذکر شد، پورت غیر فعال می‌شود (با فعال کردن Port Security، حالت shutdown به صورت پیش‌فرض فعال می‌شود). حالت Restrict بسته‌ی ارسالی با شرایط ذکر شده را بلاک کرده و این رویداد را نیز ثبت می‌کند (log). حالت protect نیز بدون ثبت رویداد، بسته را بلاک می‌کند.

نکته: در پورت‌سکیوریتی، آدرس‌های فیزیکی که به صورت ایستا برای آنها تنظیم شده باشند، به صورت پیش‌فرض دارای **aging time** نیستند؛ اما برای حالات یادگیرنده فیلد **age** مشخص می‌شود.

هر رکورد در جدول یک زمان انقضا دارد که پس از آن اگر mac روی پورت نباشد، حذف خواهد شد؛ به این امر age شدن می‌گوییم.

برای مقابله با flood شدن فریم‌هایی که مقصد تک‌پخشی یا چندپخشی ناشناخته‌ای دارند، می‌توان این بسته‌ها را با دستورهای زیر برای یک پورت مشخص بلاک کرد.

```
Switch(config-if)# switchport block unicast
Switch(config-if)# switchport block multicast
```

حملات VLAN

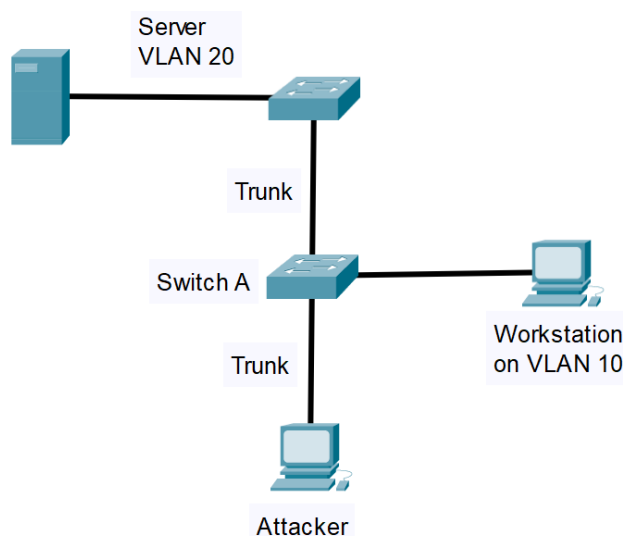
حملات VLAN، شبکه‌های محلی مجازی (VLAN) را هدف قرار می‌دهند. این حملات از آسیب‌پذیری‌های موجود در تنظیمات VLAN برای دسترسی غیرمجاز به منابع شبکه و اطلاعات شبکه‌های محلی مجازی دیگر استفاده می‌کنند. دو نمونه از انواع رایج حملات VLAN عبارتند از Switch Spoofing و Double Tagging.

Switch Spoofing

این حمله در حالتی رخ می‌دهد که Access یا Trunk بودن یک پورت با استفاده از Dynamic Trunking Protocol یا DTP تنظیم شود. در این حالت، Access یا Trunk بودن لینک به صورت پویا مشخص خواهد شد. به عبارتی، اگر بسته‌های چند شبکه محلی مجازی روی لینک حرکت کند، لینک به صورت پویا به حالت Trunk تبدیل می‌شود.

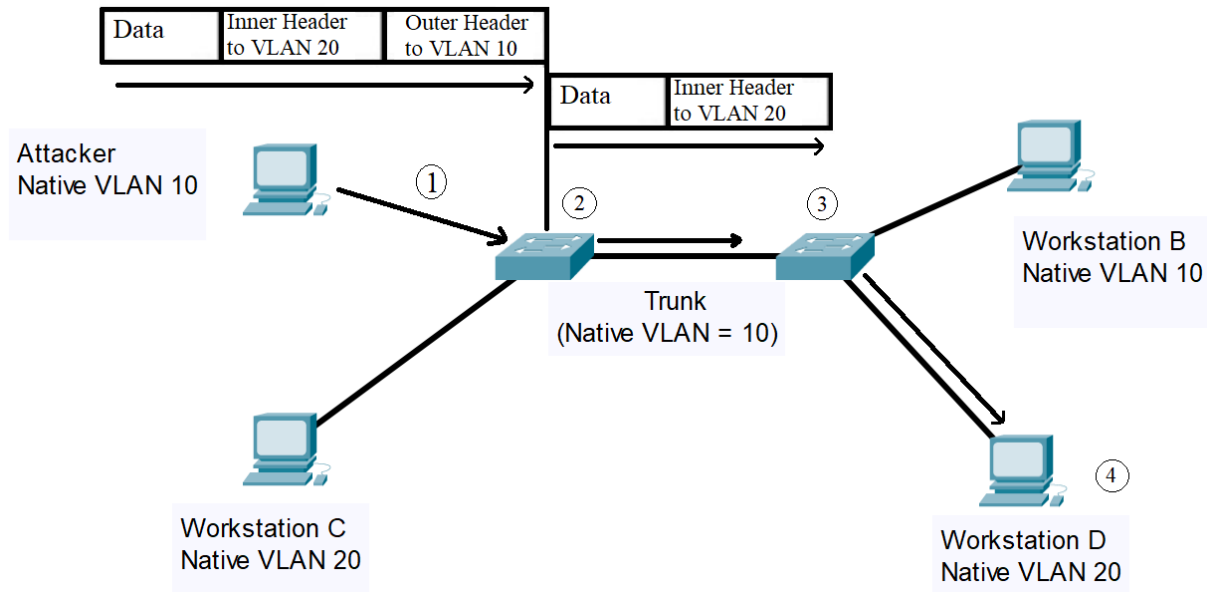
به منظور سوء استفاده، مهاجم می‌تواند پروتکل DTP را اجرا کرده و لینک بین سوئیچ و مهاجم که در حالت عادی از نوع Access است، به حالت Trunk تبدیل می‌شود. با Trunk شدن لینک، و با توجه به این که در پیکربندی اولیه، مدیر شبکه به دلیل access بودن این لینک لیست allowed vlan را مشخص نکرده است، تمامی شبکه‌های مجازی روی این لینک امکان عبور خواهند داشت و مهاجم به این ترتیب می‌تواند به تمامی شبکه‌های محلی مجازی دسترسی غیر مجاز داشته باشد.

شکل زیر به خوبی این مسئله را نشان می‌دهد. همانطور که در این شکل مشخص است، مهاجم توانسته لینک بین سوئیچ A و خودش را با سوء استفاده از DTP به حالت Trunk ببرد.



Double Tagging

در این حمله، مهاجم بسته‌ای را با دو تگ VLAN ارسال می‌کند و از این طریق اقدامات امنیتی را دور می‌زند و به داده‌های حساس دسترسی پیدا می‌کند. به عنوان مثال، شکل زیر را در نظر بگیرید.



طبق شکل بالا، مهاجم طی قدم‌های ۱ تا ۴ این حمله را انجام می‌دهد. طبق این شکل، مهاجم در Native VLAN 10 قرار دارد. مهاجم بسته‌ای روی شبکه ارسال می‌کند به صورتی که دو تگ را به بسته‌ی ارسالی اضافه می‌کند، ابتدا تگ مربوط به مقصد، و سپس بر روی آن تگ مربوط به Native VLAN خودش را قرار می‌دهد. در این صورت، هنگامی که این بسته به سوئیچ می‌رسد، سوئیچ ابتدا تگ مربوط به Native VLAN را (تگ بیرونی) حذف کرده و سپس با توجه به تگ دوم، بسته را به سمت VLAN مربوطه هدایت می‌کند. از این طریق مهاجم می‌تواند برای شبکه‌هایی که عضو آن نیست به صورت غیر مجاز بسته‌ای را ارسال کند.

برای دفاع در برابر این حمله، باید پیش‌فرض Native VLAN را تغییر داد (مثلاً Native VLAN را به 60 برای همه سوئیچ‌ها تغییر داد) و سپس آن را به طور کلی خاموش کرد و آن را به هیچ دستگاهی نسبت نداد.

دفاع در برابر حملات VLAN

برای دفاع در برابر این حملات به نکات زیر مشخص کنید:

- همه ی پورتهایی که به کاربران انتهایی (End Users) داده می شود را به صورت دستی در حالت Access قرار دهید.
- پورتهایی بلااستفاده را خاموش کنید و همه ی آن ها را عضو یک VLAN بلااستفاده کنید. به این شبکه محلی مجازی سیاهچاله (Black Hole) گفته می شود.
- روی پورتهای که به صورت Trunk تعریف شده اند، Native VLAN پیش فرض را تغییر دهید (مثلا آن را برابر با VLAN 60 قرار دهید) و این VLAN را خاموش کنید.
- روی همه ی پورتهای Trunk، به طور صریح شبکه های محلی مجازی که مجاز به عبور هستند (Allowed VLAN) را مشخص کنید.

لیست کنترل دسترسی

لیست های کنترل دسترسی (Access Control List یا ACL) مجموعه ای از قواعد هستند که تعیین می کنند کدام بسته ها مجاز به عبور از سوئیچ هستند و کدام یک باید مسدود شوند. این لیست ها می توانند ترافیک ورودی را بر اساس معیارهای مختلفی مانند آدرس IP مبدا و مقصد، نوع پروتکل، شماره پورت و موارد دیگر فیلتر کنند و برای جلوگیری از دسترسی غیرمجاز به داده های حساس، مسدود کردن ترافیک مخرب و اجرای سیاست های شبکه استفاده شوند. به طور کلی، لیست های کنترل دسترسی یک لایه امنیتی و کنترل اضافی در سوئیچ های سیسکو ارائه می دهند و به مدیران شبکه کمک می کنند تا شبکه های خود را بهتر مدیریت کنند و در برابر تهدیدات احتمالی محافظت کنند.

یک دسته بندی ممکن برای لیست های کنترل دسترسی به صورت زیر می باشد:

- Routed ACL (RACL): روی رابط های مجازی مسیریاب ها تنظیم می شوند.
 - Port ACL (PACL): روی پورت ها اعمال می شوند. این قواعد روی پورت های لایه ۲ عمل می کنند ولی می توانند از اطلاعات لایه ۳ و ۴ هم استفاده کنند.
- برای قواعد ACL سه مولفه اصلی باید تعریف شود:
- پروتکل: قاعده روی ترافیک مربوط به چه پروتکلی اعمال می شود؟ مثلا IP، IPX و ...
 - پورت: قاعده روی ترافیک مربوط به چه پورتهای اعمال می شود؟ مثلا Fast Ethernet 0/0

- قاعده روی ترافیک ورودی اعمال می شود یا ترافیک خروجی (IN یا OUT)؟
در کنار دسته بندی ذکر شده، لیست های کنترل دسترسی می توانند به صورت استاندارد یا گسترده (Extended) باشند. در حالت گسترده، قواعد لیست کنترل دسترسی می تواند با توجه به مبدا آدرس IP و مقصد آدرس IP تعریف شود ولی در حالت استاندارد قواعد فقط با توجه به آدرس مبدا نوشته می شوند.

دستور مربوط به ساخت یک لیست کنترل دسترسی استاندارد

```
Switch(config)# access-list [1-99 | 1300-1999] [permit | deny | remark] source
```

در این دستور، اعداد ۱ تا ۹۹ و ۱۳۰۰ تا ۱۹۹۹، یک شناسه برای لیست ساخته شده هستند. لیست های استاندارد می توانند شماره های ۱ تا ۹۹ و ۱۳۰۰ تا ۱۹۹۹ را به عنوان شناسه داشته باشند. هر قاعده در لیست کنترل دسترسی می تواند در یکی از ۳ دسته اعمال permit، deny یا remark تعریف شود. Permit به این معناست که در صورتی که بسته ای با آدرس IP مشخص شده دریافت شد اجازه ی عبور داشته باشد در حالی که Deny بسته را بلاک می کند. Remark نیز برای اضافه کردن توضیحات استفاده می شود.

دستور مربوط به ساخت یک لیست کنترل دسترسی گسترده

```
Switch(config)# access-list [100-199 | 2000-2699] protocol [ip | tcp | ...]  
[permit | deny | remark] [source] [destination]
```

- در لیست های گسترده، علاوه بر آدرس مبدا، آدرس مقصد و پروتکل مورد نظر را نیز می توان در قاعده مشخص کرد.
- دقت کنید که در دستورات بالا، source و destination می توانند ۳ حالت داشته باشند:
- any: قاعده برای هر آدرسی اعمال می شود

- host A.B.C.D: قاعده فقط برای آدرس یک میزبان که برابر با A.B.C.D است اعمال می شود
- A.B.C.D Wildcard Mask: قاعده برای تمام آدرس های حاصل از اعمال wildcard mask روی آدرس A.B.C.D اعمال می شود. توضیحات مربوط به Wildcard Mask در ادامه ی پیش گزارش آورده شده است.

دستور حذف لیست کنترل دسترسی

با دستور زیر می توان یک لیست کنترل دسترسی را حذف کرد:

```
Switch(config)# no access-list [1-199 | 1300 – 2699]
```

لیست کنترل دسترسی نامگذاری شده

لیست های کنترل دسترسی را می توان بر اساس شناسه ای که دارند به دو دسته ی نامگذاری شده (Named) و شماره گذاری شده (Numbered) تقسیم کرد. در دستورات قبلی، از لیست کنترل دسترسی شماره گذاری شده استفاده شد. لیست های نامگذاری شده به صورت زیر تعریف می شود.

```
Switch(config)# ip access-list [extended | standard] name
```

بعد از وارد کردن این دستور، می توان یک لیستی از قواعد را به صورت زیر به این لیست که با نام name ساخته شد اضافه کرد.
برای لیست های نامگذاری شده ی استاندارد:

```
Switch (config-std-nacl)# [permit | deny | remark] source
```

برای لیست‌های نامگذاری شده‌ی گسترده:

```
Switch (config-ext-nacl)# [permit | deny | remark] protocol source destination
```

مثال:

```
Switch (config-std-nacl)# permit any 172.16.1.0 0.0.0.255  
Switch (config-ext-nacl)# deny ip any 172.16.1.0 0.0.0.255
```

مزیت لیست‌های نامگذاری شده این است که علاوه بر داشتن یک نام با مفهوم برای لیست کنترل دسترسی، قابل اصلاح نیز هستند و می‌توان در ترتیب و قواعد کنترلی در لیست کنترل دسترسی مورد نظر تغییراتی اعمالی کرد. ولی در ACL‌های شماره‌گذاری شده، قواعد فقط به انتهای لیست اضافه می‌شوند و اگر به فرض بخواهیم بین دو قاعده یک قاعده جدید اضافه کنیم، باید کل قواعد بعد از قاعده اول را حذف کنیم، قاعده جدید را اضافه و سپس دوباره قواعد حذف شده را اضافه کنیم. برای حذف یک قاعده در لیست‌های شماره‌گذاری شده از دستور زیر استفاده می‌شود.

```
Switch (config)# no access-list number [permit | deny | remark] source
```

در نهایت برای اینکه قواعد مربوط به یک لیست کنترل دسترسی، بر روی یک رابط اعمال شود باید به صورت زیر آن را به رابط مورد نظر تخصیص داد.

```
Switch (config)# interface [interface_name]  
Switch (config-if)# ip access-group [num | name] [in | out]
```

نکته: به ترتیب قواعدی که در لیست کنترل دسترسی اضافه می‌کنید دقت کنید. هنگام دریافت یک بسته، قواعد لیست کنترل دسترسی به ترتیب از ابتدا به انتها و یک به یک برای بسته‌ی دریافت شده بررسی می‌شود و اولین قاعده‌ای که با شرایط بسته دریافتی مطابقت داشته باشد روی بسته اعمال می‌شود و قواعد بعدی در لیست بررسی نمی‌شوند. به عنوان مثال: اگر روی رابط Fa0/1 لیست کنترل دسترسی با دو قاعده روبرو را داشته باشیم.

```
permit any
deny 172.16.0.0 0.0.255.255
```

همه‌ی بسته‌های دریافتی بر روی Fa0/1 با قاعده اول مطابقت خواهند داشت و همه‌ی بسته‌ها اجازه‌ی عبور خواهند داشت و هیچگاه قاعده دوم بررسی نمی‌شود.

نکته: در انتهای همه‌ی لیست‌های کنترل دسترسی، یک قاعده به صورت ضمنی وجود دارد که همه‌ی بسته‌ها را بلاک می‌کند (یک قاعده deny). در حقیقت، این قاعده ضمنی باعث می‌شود که هر بسته‌ای که با قواعد اضافه شده توسط کاربر به لیست match نشود، دراپ شود.

Wildcard Mask

Wildcard Masking در دستورات تعریف لیست کنترل دسترسی به منظور مشخص کردن محدوده آدرس‌های IP مبدا یا مقصد استفاده می‌شود. این Mask از ۳۲ بیت تشکیل می‌شود که هر بیت 0 به معنای لزوم مطابقت بیت متناظر آدرس‌های حاصل و آدرس وارد شده در دستور می‌باشد و بیت 1 به معنای آزاد بودن بیت متناظر است. به عنوان مثال، دستور زیر را در نظر بگیرید.

```
Switch (config-if)# permit ip any 192.168.1.0 0.0.254.255
```

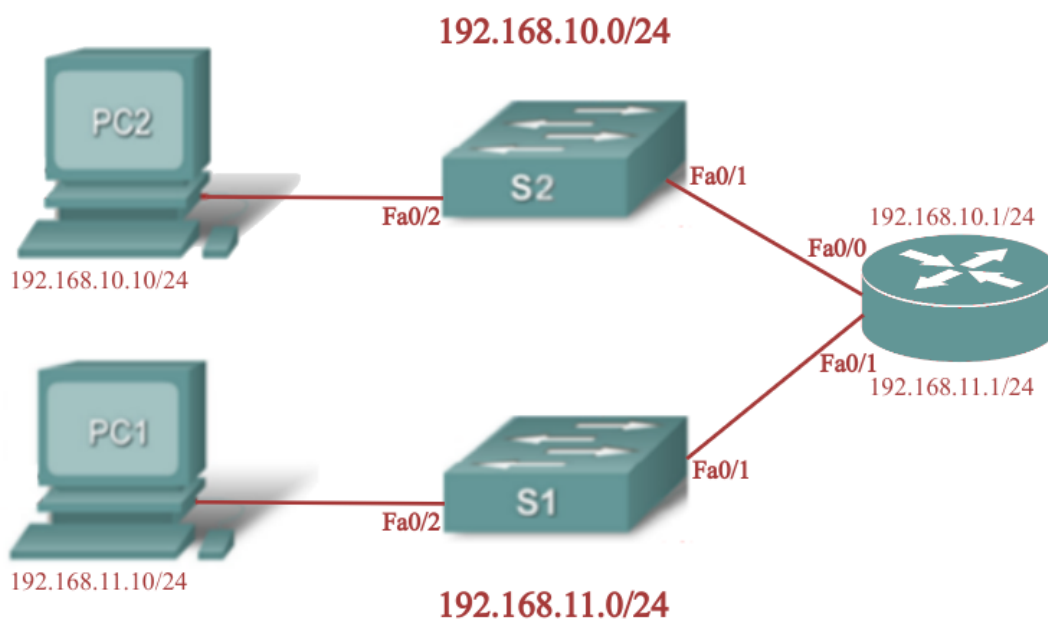
در این دستور، wildcard mask برابر با 0.0.254.255 است که معادل آن در مبنای دودویی برابر با 00000000.00000000.11111110.11111111 خواهد بود. چون دو بایت اول تماماً صفر هستند، دو

بایت اول تمام آدرس‌های حاصل برابر با دو بایت اول آدرس شده، یعنی به ترتیب 192 و 168 خواهند بود. همچنین برای بایت سوم، بیت انتهایی آدرس وارد شده عیناً در خروجی تکرار خواهد شد و بقیه‌ی بیت‌ها می‌توانند 0 یا 1 باشند. همه‌ی بیت‌های بایت چهارم نیز می‌توانند 0 یا 1 باشند. بنابراین آدرس‌های نهایی به صورت 192.168.1.0/24، 192.168.3.0/24، 192.168.5.0/24 تا 192.168.255.0/24 خواهند بود.

نکته: wildcard mask برای آدرس یک میزبان برابر با 0.0.0.0 خواهد بود.

مثال‌هایی از استفاده از ACL

در ادامه، دو مثال کاربردی از نحوه استفاده از لیست‌های کنترل دسترسی را بررسی می‌کنیم. فرض کنید شبکه‌ای به شکل زیر داریم.



می‌خواهیم به کمک قواعد لیست دسترسی، از عبور ترافیک PC2 با آدرس 192.168.11.10 به سمت شبکه 192.168.10.0/24 و PC1 جلوگیری کنیم. دقت کنید که می‌خواهیم این محدودیت فقط بر روی PC2 اعمال شود و بقیه‌ی دستگاه‌های موجود در 192.168.11.0/24 شامل این محدودیت نمی‌شوند. در ادامه چگونگی اعمال این محدودیت به کمک لیست‌های کنترل دسترسی استاندارد شماره‌گذاری شده و نامگذاری شده توضیح داده می‌شود.

اعمال سیاست به کمک لیست کنترل دسترسی استاندارد نامگذاری شده

برای اعمال سیاست، ابتدا بر روی روتر R1 این لیست را با نام NO_ACCESS تعریف می‌کنیم و قواعد زیر را به این لیست اضافه می‌کنیم.

- (1) deny host 192.168.11.10
- (2) permit 192.168.11.0 0.0.0.255

قاعده اول باعث می‌شود هر بسته‌ای با آدرس مبدا 192.168.11.10 دراپ شود. همچنین، به یاد داشته باشید که برای هر لیست کنترل دسترسی، از جمله لیست NO_ACCESS که در اینجا تعریف شده است، همواره یک قاعده ضمنی در انتهای لیست قواعد وجود دارد که به صورت پیشفرض همه‌ی بسته‌ها را دراپ می‌کند. بنابراین افزودن قاعده دوم در اینجا برای اجازه دادن به عبور بسته‌هایی که از دستگاه‌های دیگر از شبکه 192.168.11.0/24 ارسال می‌شوند ضروری است.

پس از ساختن لیست NO_ACCESS و اضافه کردن دو قاعده ذکر شده، باید آن را بر روی پورت Fa0/0 روتر R1 اعمال کرد. بدون اعمال قواعد کنترل دسترسی به یک پورت، این قواعد هیچ تاثیری نخواهند داشت. به این منظور، ابتدا با دستور زیر وارد پیکربندی Fa0/1 می‌شویم.

```
Router(config-std-nacl)# interface Fa0/0
```

سپس به کمک دستور زیر، برای همه‌ی بسته‌هایی که از این پورت خارج می‌شوند (out) این قواعد را اعمال می‌کنیم. به این صورت از خروج بسته‌های ارسال شده از مبدا PC2 به سمت PC1 جلوگیری می‌شود.

```
Router(config-if)# ip access-group NO_ACCESS out
```

دستورات مربوط به این مثال به شکل زیر خواهد بود.

```
R1#show access-lists
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#ip access-list standard NO_ACCESS
R1(config-std-nacl)#deny host 192.168.11.10
R1(config-std-nacl)#permit 192.168.11.0 0.0.0.255
R1(config-std-nacl)#interface Fa0/0
R1(config-if)#ip access-group NO_ACCESS out
R1(config-if)#do show access-lists
Standard IP access list NO_ACCESS
 10 deny host 192.168.11.10
 20 permit 192.168.11.0 0.0.0.255
```

همانطور که مشخص است، در نهایت یک لیست کنترل دسترسی با نام NO_ACCESS لیست و با دو قاعده ذکر شده در بالا در روتر R1 ذخیره شده است. در لیست NO_ACCESS، دو قاعده ذکر شده به ترتیب با اندیس‌های 10 و 20 به لیست اضافه شده‌اند. فرض کنید که می‌خواهیم علاوه بر آدرس 192.168.11.10، برای آدرس 192.168.11.11 نیز یک قاعده deny بنویسیم. این قاعده باید به ابتدای لیست و یا بین قاعده deny و permit اضافه شود، در غیر این صورت هیچ تاثیری نخواهد داشت. برای اضافه کردن یک قاعده به لیست‌ها نامگذاری شده‌ی بالا می‌توان به صورت زیر عمل کرد.

```
R1(config)#ip access-list standard NO_ACCESS
R1(config-std-nacl)#11 deny host 192.168.11.11
R1(config-std-nacl)#exit
R1(config)#do show access-lists
Standard IP access list NO_ACCESS
 10 deny host 192.168.11.10
 11 deny host 192.168.11.11
 20 permit 192.168.11.0 0.0.0.255
```

همانطور که مشخص است، پس از وارد شدن به محیط پیکربندی لیست NO_ACCESS با دستور زیر قاعده مورد نظر را در اندیس 11 به لیست اضافه کردیم.

```
Router(config-std-nacl)# 11 deny host 192.168.11.11
```

اعمال سیاست به کمک لیست کنترل دسترسی استاندارد شماره‌گذاری شده در این بخش، برای مثال قبلی، یک لیست کنترل دسترسی استاندارد شماره‌گذاری شده می‌سازیم. به این منظور، از دستورات زیر استفاده می‌کنیم.

```
Router(config)# access-list 10 deny 192.168.11.10
Router(config)# access-list 10 permit 192.168.11.0 0.0.0.255
```

با اجرای دستورات بالا، یک لیست کنترل دسترسی شماره‌گذاری شده با شناسه 10 خواهیم داشت که دو قاعده deny و permit به ترتیبی که نوشته شده‌اند در این لیست قرار گرفته‌اند.

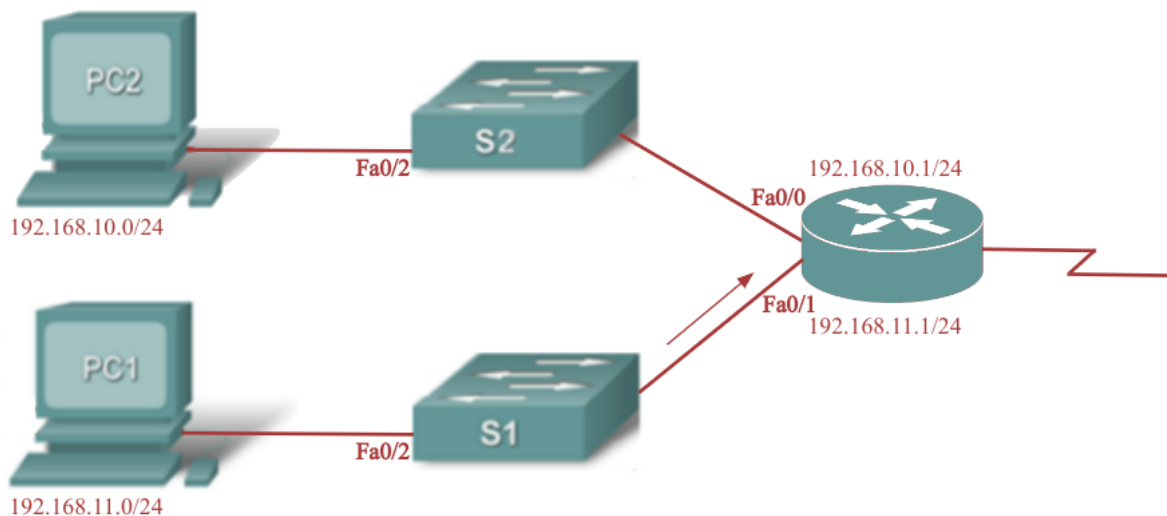
```

R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 10 deny 192.168.11.10
R1(config)#access-list 10 permit 192.168.11.0 0.0.0.255
R1(config)#do show access-lists
Standard IP access list NO_ACCESS
 10 deny host 192.168.11.10
 20 permit 192.168.11.0 0.0.0.255
Standard IP access list 10
 10 deny host 192.168.11.10
 20 permit 192.168.11.0 0.0.0.255

R1(config)#interface Fa0/0
R1(config-if)#ip access-group 10 out

```

در مثال دوم، فرض کنید شبکه‌ای به شکل زیر داریم.



در این شبکه می‌خواهیم به کمک لیست‌های گسترده، از اتصال FTP از 192.168.11.0/24 به 192.168.10.0/24 جلوگیری کنیم. به این منظور، یک لیست کنترلی دسترسی گسترده برای deny کردن بسته‌هایی که در ترافیک ورودی به Fa0/1 قرار دارند و از مبدا 192.168.11.0/24 به مقصد 192.168.10.0/24 و پورت 20 یا 21 می‌روند، تعریف می‌کنیم. در این لیست کنترل دسترسی، همچنین می‌خواهیم ترافیک‌های IP بین هیچ محدودیتی و از هر مبدا به هر مقصدی عبور کنند. این کار را ابتدا به کمک لیست‌های گسترده شماره‌گذاری شده انجام می‌دهیم.

اعمال سیاست به کمک لیست کنترل دسترسی گسترده شماره گذاری شده:

برای اعمال سیاست مورد نظر، باید یک لیست کنترل دسترسی با قواعد زیر داشته باشیم.

(1) deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 21

(2) deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 20

(3) permit ip any any

قواعد 1 و 2، بسته های TCP که از مبدا 192.168.11.0/24 و به مقصد 192.168.10.0/24 و پورت های 21 و 20 (پورت های مربوط به پروتکل FTP) ارسال می شوند را دراپ می کنند و قاعده سوم اجازه عبور به ترافیک IP بین هر مبدا و مقصدی را می دهد. مطابق دستورات زیر، این قواعد را در لیست کنترل دسترسی گسترده شماره 101 اضافه می کنیم و سپس آن را در پورت Fa0/1 برای ترافیک های ورودی (in) اعمال می کنیم.

```
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 21
R1(config)#access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 20
R1(config)#access-list 101 permit ip any any
R1(config)#do show access-lists
Extended IP access list 101
 10 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq ftp
 20 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 20
 30 permit ip any any

R1(config)#interface Fa0/1
R1(config-if)#ip access-group 101 in
```


اعمال سیاست به کمک لیست کنترل دسترسی گسترده نامگذاری شده شده:

سیاست مورد نظر را به صورت زیر می توان در قالب لیست های نامگذاری شده نوشت.

```
R1(config)#ip access-list extended DENY_FTP
R1(config-ext-nacl)#deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 21
R1(config-ext-nacl)#deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 20
R1(config-ext-nacl)#permit ip any any
R1(config-ext-nacl)#do show access-lists
Extended IP access list 101
 10 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq ftp
 20 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 20
 30 permit ip any any
Extended IP access list DENY_FTP
 10 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq ftp
 20 deny tcp 192.168.11.0 0.0.0.255 192.168.10.0 0.0.0.255 eq 20
 30 permit ip any any
```

امنیت مدیریت سوئیچ ها به کمک ACL

یکی از کاربردهای ACL، محدود کردن دسترسی به رابط ترمینال سوئیچهای شبکه است. در سوئیچها برای تنظیم رابط مربوط به اتصال به ترمینال از دستور line vty استفاده می شود. از این خطوط برای اتصال SSH یا Telnet به سوئیچ استفاده می شود. به عنوان مثال، به کمک دستور زیر، برای خطوط 0 تا 4 ابتدا یک رمز عبور و سپس لیست کنترلی شماره ۲۱ روی ترافیک ورودی اعمال می شود.



```
Switch (config)# line vty 0 4
Switch (config-line)# password secret
Switch (config-line)# login
Switch (config-line)# access-class 21 in
```

سربلند باشید