iptables 명령

룰셋(rule-sets) 저장 및 복원

iptables는 룰셋을 특정 파일 포맷으로 저장하고 복원하는 iptables-save, iptables-restore 명령이 있다.
iptables-restore의 stdin(표준입력)으로 복원이 가능하다
다음과 같은 명령을 실행하면 된다. make-rules.sh | iptables-restore

```
iptables-save [-c] [-t table]
```

-c : 지정된 바이트와 패킷 카운터 값을 유지하는 옵션, 기본 방화벽을 재부팅(reboot) 하려는 경우에 유용하다. 통계 및 회계 루틴을 중단하지 않고 재부팅할 수 있다.
-t : 저장할 테이블을 지정한다. 생략하면 모든 테이블을 자동으로 저장한다.

```
iptables-restore [-c] [-n]
```

-c : 이전에 iptables-save로 저장된 카운터를 복원하려는 경우 바이트 및 패킷 카운터를 복원하고 싶으면 꼭 써야하는 옵션이다.
-n : iptables-restore가 쓰고 있는 테이블에 이전에 작성된 규칙을 덮어쓰지 않도록 지시한다.
iptables-restore의 기본 동작은 이전에 삽입된 모든 규칙들을 플러시(flush)하고 파기(destory)한다.

iptables 명령1

각각 Rule은 커널이 패킷으로 무엇을 할지 알아내기 위해 살펴보는 라인(line)이다.
모든 기준 또는 일치 항목이 충족되면 target 또는 jump 명령을 수행한다. 명령 형식은 다음과 같다.

```
iptables [-t table] command [match] [target/jump]
iptables [-t《table-name》] command《chain-name》《parameter-N》《option-N》
-t : 테이블 지정 없으면 기본값은 filter 테이블이다.
chain-name 뒤에 파라미터와 옵션의 쌍이 여러개 올 수 있다.
```

Commands 옵션	*예제 / 설명*
-A, --append	*iptables -A INPUT ...* *체인의 끝에 규칙(Rule)을 추가한다. 즉, 규칙 집합(rule-set)에서 마지막에 체크한다.*
-D, --delete	*iptables -D INPUT --dport 80 -j DROP* *iptables -D INPUT -s 221.194.47.0/24 -j REJECT* *iptables -D INPUT 1* *체인에서 규칙을 삭제한다.* *방법1)패킷에 일치시킬 전체 규칙을 입력 또는 방법2)일치시킬 규칙 번호를 지정해서 삭제할 수 있다.* *방법1)은 항목이 체인의 항목과 정확히 일치해야하고 방법2)는 규칙 번호와 일치해야 한다.* *(규칙 번호는 1부터 시작하여 각 체인의 상단부터 번호가 매겨진다.)*
-R, --replace	*iptables -R INPUT 1 -s 192.168.0.1 -j DROP* *지정된 라인에서 이전의 항목을 바꾼다.*
-I, --insert	*iptables -I INPUT 1 --dport 80 -j ACCEPT* *iptables -I INPUT 1 -s 59.45.175.10 -j ACCEPT* *체인의 지정된 라인에 규칙을 삽입한다.*
-L, --list	*iptables –L —line–numbers* *iptables -L INPUT* *iptables -t nat -L* *iptables -t mangle -L* *지정된 체인의 모든 항목을 보여준다. 체인을 지정하지 않아도 된다.*
-F, --flush	*iptables -F INPUT* *iptables -t nat -F* *지정된 체인의 모든 규칙을 플러시한다(하나씩 전부다 삭제하는 것과 같다)* *체인을 지정하지 않으면 지정된 테이블의 모든 체인의 모든 규칙을 다 삭제한다.*
-Z, --zero	*iptables -Z INPUT* *특정 체인(지정하지 않으면 모든 체인) 모든 카운터를 0으로 만들도록 지시한다.* *-L, -v 2개의 옵션으로 패킷 카운터를 볼 수 있다*
-N, --new-chain	*iptables -N mychain* *지정된 테이블에 지정된 이름의 체인을 생성*
-X, --delete-chain	*iptables -X mychain* *테이블에서 지정된 체인을 삭제(삭제할 체인에 규칙이 아무것도 없어야 한다)* *빌트인(built-in) 체인은 삭제할 수 없다. 체인이름을 지정하지 않으면 빌트인 체인을 제외한 나머지 체인을 모두 삭제한다.*
-P, --policy	*iptables -P INPUT DROP* *지정된 체인의 기본 정책(policy) 또는 타겟(target)을 설정한다.* *패킷이 모든 규칙(rule)과 일치하지 않고 체인을 통과할 때 지정된 target으로 전송된다*

Commands 옵션	예제 / 설명
-E, --rename-chain	iptables -E oldchain newchain 체인의 이름을 변경한다. (테이블의 작동에는 영향을 주지 않는다)

PARAMETERS	설명
-m, --match <match>	사용할 일치 항목, 즉 특정 속성을 테스트하는 확장 모듈을 지정합니다. 일치 항목 세트는 target이 호출되는 조건을 구성합니다. 일치 항목들은 명령줄에 지정된 대로 처음부터 끝까지 평가되며 short-circuit 방식으로 작동합니다. 즉, 하나의 확장이 false를 생성하면 평가가 중지됩니다.
-j, --jump <target>	규칙의 target을 지정합니다. 즉, 패킷이 일치하면 어떻게 할지 운명을 결정합니다. target은 이 규칙을 포함하는 체인이 아닌 또다른 사용자 정의 체인, 패킷의 운명을 즉시 결정하는 특수 내장 target, 또는 EXTENSIONS 일 수 있습니다.
-g, --goto <chain>	이는 사용자가 지정한 체인에서 처리가 계속되어야 함을 뜻합니다.

---

iptables 명령2 - Iptables matches

iptable 일치에 대해 더 자세히 보면 5개의 다른 카테고리로 요약할 수 있다.
1. 모든 규칙에서 사용할 수 있는 일반 일치(generic matches)
2. TCP 패킷에만 적용할 수 있는 TCP 일치(TCP matches)
3. UDP 패킷에만 적용할 수 있는 UDP 일치(UDP matches)
4. ICMP 패킷에만 적용할 수 있는 ICMP 일치(ICMP matches)
5. 상태(state), 소유자(owner), 및 제한 일치(limit matches)와 같은 특수 일치(special matches)

Generic matches

옵션	예제 / 설명
-p, --protocol	iptables -A INPUT -p tcp iptables -A INPUT -p tcp -s 1.2.3.4 --dport 80 -j DROP iptables -A INPUT -p icmp -i eth0 -j DROP 지정된 프로토콜은 tcp, udp, udplite, icmp, icmpv6,esp, ah, sctp, mh 또는 특수 키워드 all 중 하나이거나 앞서 말한 프로토콜 중 하나 또는 다른 프로토콜을 나타내는 숫자 값일 수 있다. 예를 들어 ICMP는 1, TCP는 6, UDP는 17 /etc/protocols의 프로토콜 이름도 허용된다. 숫자 0은 all과 동일합니다. all은 모든 프로토콜과 일치하며 해당 옵션이 생략되면 기본값으로 사용됩니다.
-s, --src, --source	iptables -A INPUT -s 192.16.22.41,192.16.22.43 -p icmp -j REJECT iptables -t filter -A INPUT -s 59.45.175.0/24 -j REJECT (59.45.175.0부터 59.45.175.255까지의 모든 IP를 차단) Source IP 기반으로 패킷을 일치시킨다. '/24'와 같은 CIDR notation을 추가하여 255.255.255.0 와 같은 넷마스크를 사용할 수 있다. 예를 들어 192.168.0.0/24 로 지정하면 192.168.0.x 범위의 모든 패킷과 일치
-d, --dst, --destination	iptables -A OUTPUT -d 192.168.1.0/24 -j DROP Destination IP 기반으로 패킷을 일치시킨다. 192.168.0.0/255.255.255.0 또는 192.168.0.0/24 이렇게 범위지정도 가능하다
-i, --in-interface	iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j LOG --log-prefix "IP_SPOOF A: " iptables -A INPUT -i eth1 -p tcp --dport 80 -j DROP iptables -t nat -A PREROUTING -s 192.168.1.2 -i eth0 -j MASQUERADE iptables -t nat -A PREROUTING -i wlan0 -d 10.1.1.7 -j DNAT --to-destination 192.168.1.2 일치하는 패킷이 들어온 인터페이스에 사용된다. 따라서 INPUT, FORWARD, PREROUTING 체인에서만 유효하며 다른 곳에서 사용하면 오류 메시지를 반환 특정 인터페이스를 지정하지 않으면 기본값은 문자열 '+' 값인데 모든 인터페이스에 들어오는 패킷을 일치시킨다. '+'는 인터페이스 유형에 붙여서 쓸 수 있다. 즉 eth+ 는 모든 이더넷 장치가 된다.
-o, --out-interface	iptables -A FORWARD -o eth0 iptables -A OUTPUT -o eth1 -d 192.168.1.0/24 -j DROP iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 192.168.1.1 (eth0에서 나가는 모든 패킷의 소스 IP는 192.168.1.1입니다.) iptables -t nat -A POSTROUTING -o wlan0 -s 192.168.1.2 -p udp --dport 16020 -j SNAT --to 10.1.1.7:51889 iptables -t nat -A POSTROUTING -o wlan0 -s 192.168.1.2 -p tcp --dport 21 -j SNAT --to 10.1.1.7:21 iptables -t nat -A POSTROUTING -o wlan0 -s 192.168.1.3 -j SNAT --to 10.1.1.9 iptables -t nat -A POSTROUTING -o eth1 -s 192.168.1.22 -p all -j SNAT --to 192.168.20.1 인터페이스로 떠나는 패킷에 사용된다. 따라서 OUTPUT, FORWARD 및 POSTROUTING 체인에서만 사용할 수 있다. 특정 인터페이스를 지정하지 않으면 패킷이 어디로 가는지 상관없이 모든 장치를 일치시킨다.
-f, --fragment	iptables -A INPUT -f 조각난 패킷에 사용

TCP matches

옵션	예제 / 설명
--sport, --source-port	iptables -A INPUT -p tcp --sport 22 출발지 포트(source port) 로 패킷을 일치시킨다 서비스(서비스 이름은 /etc/services 파일에 있어야 한다) 이름 또는 포트를 기반으로 일치시킨다. 포트의 범위를 지정할 수도 있다. 예를 들어 --source-port 22:80 로 하면 22와 80 사이의 모든 포트와 일치 --source-port [p1:p2] 형식에서 p1과 p2는 생략시 각각 0과 65535로 간주된다.
--dport, --destination-port	iptables -A INPUT -p tcp --dport 22 iptables -A INPUT -p tcp -m tcp --dport 22 -s 59.45.175.0/24 -j DROP iptables -A INPUT -p tcp -m multiport --dports 22,5901 -s 59.45.175.0/24 -j DROP iptables -A INPUT -p tcp -m multiport ! --dports 22,80,443 -j DROP ('!' 는 Negation operator) (tcp 모듈로는 여러 포트를 지정할 수 없지만 multiport 모듈로는 지정할 수 있다.)

옵션	예제 / 설명
	##### block outgoing traffic - specific port (append rules) iptables -t filter -A OUTPUT -p tcp -m tcp --dport 80 -d 192.168.0.110 -j DROP # drops connections comming from your current host to 192.168.0.110:80 ##### block outgoing traffic - multiple ports (append rules) iptables -t filter -A OUTPUT -p tcp -m multiport --dports 80,200 -d 192.168.0.0/24 -j DROP # drops connections comming from your current host to 192.168.0.0/24, port 80 & 200 iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080 (이 규칙은 포트 80에서 들어오는 모든 요청을 포트 8080으로 전달) 목적지 포트에 따라 TCP 패킷을 일치시킨다. 위에 설명한 --source-port 옵션과 기능만 다를 뿐 정확이 동일한 구문을 사용한다. 이 일치는 여러개의 분리된 포트 및 포트 범위를 처리하지 않는다. (자세한 내용은 multiport match extension를 참조)
--tcp-flags	iptables -p tcp --tcp-flags SYN,FIN,ACK SYN iptables -A INPUT -p tcp --tcp-flags ALL SYN,FIN -j DROP iptables -A INPUT -p tcp -m tcp --tcp-flags SYN,FIN SYN,FIN -j DROP iptables -A INPUT -p tcp -m tcp --tcp-flags ALL FIN,PSH,URG -j DROP (위 예제에서는 전체 플래그를 확인해야 하지만 FIN,PSH,URG만 일치하도록 설정) TCP 패킷의 헤더 구조를 보면 플래그(ACK, FIN, PSH, RST, SYN, URG)를 저장하는 제어 비트가 있다. 이 옵션은 TCP 플래그로 패킷을 일치시키고, 두 개의 파라미터를 받는다. 첫 번째 파라미터는 패킷에서 검사할 프래그 목록(마스크), 두 번째 파라미터는 일치하도록 설정해야 하는 플래그 목록이다. 두 파라미터 모두 리스트를 콤마(,)로 구분해야 한다. 위 첫번째 예시는 SYN 플래그가 설정(1로 셋팅)되고 ACK 및 FIN 플래그가 설정되지 않은 TCP 패킷과만 일치합니다. ALL : 모든 플래그를 사용한다는 의미, NONE : 옵션에 플래그를 사용하지 않는다는 의미 --tcp-flags ALL NONE 은 다시말해 모든 TCP 플래그를 확인하고 플래그가 설정되지 않은 경우 일치한다는 의미이다.
--syn	iptables -p tcp --syn ipchains 시대의 오래된 유물이며 호환성을 위해 여전히 존재 패킷에 SYN 비트가 설정되어 있고 ACK 및 RST 비트가 설정되어 있지 않은 경우 패킷을 일치시킨다. 즉, 이 명령은 --tcp-flags SYN,RST,ACK SYN 일치와 정확히 동일합니다.
--tcp-option	iptables -p tcp --tcp-option 16 TCP 옵션에 따라 패킷을 일치시키는 데 사용된다. TCP 옵션 : TCP 패킷 헤더의 특정 부분이고 3개의 다른 필드로 구성된다. 첫 번째 8비트는 이 스트림에서 어떤 옵션이 사용되는지 알려준다. 두 번째 8비트는 옵션 필드의 길이를 알려준다. 세 번째는 TCP 패킷의 구조상 실제 옵션데이터로 추측된다. 비트 길이는 모른다

UDP matches

옵션	예제 / 설명
--sport, --source-port	iptables -A INPUT -p udp --sport 53 출발지 포트(source port)로 UDP 패킷을 일치시킨다. 포트의 범위 지정도 가능하다 예를 들어 --source-port 22:80 로 하면 22와 80 사이의 모든 UDP 포트와 일치 --source-port [p1:p2] 형식에서 p1과 p2는 생략시 각각 0과 65535로 간주된다.
--dport, --destination-port	iptables -A INPUT -p udp --dport 53 목적지 포트(destination port)에 따라 UDP 패킷을 일치시킨다. 위에 설명한 TCP 일치의 --dport 옵션과 정확히 동일하다 적용대상이 UDP 패킷일 뿐이다.

ICMP matches

옵션	예제 / 설명
--icmp-type	iptables -A INPUT -p icmp --icmp-type 8 iptables -A INPUT -p icmp -m icmp --icmp-type 17 -j DROP ICMP type으로 패킷을 일치시킨다. ICMP type은 숫자나 이름을 지정할 수 있다. Netfilter는 모든 ICMP 타입과 일치시키기 위해 내부적으로 ICMP type 255를 사용한다. iptables -A INPUT -p icmp --icmp-type 255 -j DROP 과 일치하도록 규칙을 설정하면 모든 ICMP 패킷이 삭제됩니다. (즉, 모든 ICMP 유형을 일치시키는 데 사용됩니다.)

---

iptables 명령3 - 기타 여러가지 matches

Addrtype Types

Type	Type 설명
ANYCAST	This is a one-to-many associative connection type, where only one of the many receiver hosts actually receives the data. This is for example implemented in DNS. You have single address to a root server, but it actually has several locations and your packet will be directed to the closest working server. Not implemented in Linux IPv4.
BLACKHOLE	A blackhole address will simply delete the packet and send no reply. It works as a black hole in space basically. This is configured in the routing tables of linux.
BROADCAST	A broadcast packet is a single packet sent to everyone in a specific network in a one-to-many relation. This is for example used in ARP resolution, where a single packet is sent out requesting information on how to reach a specific IP, and then the host that is authoritative replies with the proper MAC address of that host.
LOCAL	An address that is local to the host we are working on. 127.0.0.1 for example.
MULTICAST	A multicast packet is sent to several hosts using the shortest distance and only one packet is sent to each waypoint where it will be multiple copies for each host/router subscribing to the specific multicast address. Commonly used in one way streaming media such as video or sound.
NAT	An address that has been NAT'ed by the kernel.

Type	Type 설명
PROHIBIT	Same as blackhole except that a prohibited answer will be generated. In the IPv4 case, this means an ICMP communication prohibited (type 3, code 13) answer will be generated.
THROW	Special route in the Linux kernel. If a packet is thrown in a routing table it will behave as if no route was found in the table. In normal routing, this means that the packet will behave as if it had no route. In policy routing, another route might be found in another routing table.
UNICAST	A real routable address for a single address. The most common type of route.
UNREACHABLE	This signals an unreachable address that we do not know how to reach. The packets will be discarded and an ICMP Host unreachable (type 3, code 1) will be generated.
UNSPEC	An unspecified address that has no real meaning.
XRESOLVE	This address type is used to send route lookups to userland applications which will do the lookup for the kernel. This might be wanted to send ugly lookups to the outside of the kernel, or to have an application do lookups for you. Not implemented in Linux.

옵션	예제 / 설명
--src-type	iptables -A INPUT -m addrtype --src-type UNICAST 이 옵션은 패킷의 출발지 주소 유형을 일치시키는데 사용된다. BROADCAST,MULTICAST와 같이 콤마(,)로 구분된 여러 유형을 사용할 수 있습니다.
--dst-type	iptables -A INPUT -m addrtype --dst-type UNICAST --dst-type은 --src-type과 완전히 동일한 방식으로 작동하며 구문도 동일합니다. 유일한 차이점은 목적지 주소 유형에 따라 패킷을 일치시킨다는 것이다.

Connection Tracking Matches

옵션	예제 / 설명
--ctstate	iptables -A INPUT -p tcp -m conntrack --ctstate RELATED (-m conntrack 을 반드시 지정해야 한다) 이 옵션은 conntrack 상태에 따라 패킷의 상태를 일치시키는 데 사용된다. 이 옵션에 유효한 항목은 다음과 같다. 1. INVALID 2. ESTABLISHED 3. NEW 4. RELATED 5. SNAT 6. DNAT 위 항목들은 콤마(,)로 구분해 여러개 지정할 수 있다. 예) -m conntrack --ctstate ESTABLISHED,RELATED 옵션 앞에 느낌표(!)를 사용해 반전(invert)도 가능하다. 예) -m conntrack ! --ctstate ESTABLISHED,RELATED 이렇게 하면 ESTABLISHED, RELATED 상태를 제외한 모든 상태와 일치 예) iptables -A INPUT -p tcp -m conntrack --ctstate NEW -m tcp ! --tcp-flags FIN,SYN,RST,ACK SYN -j DROP
--ctorigsrc	iptables -A INPUT -p tcp -m conntrack --ctorigsrc 192.168.0.0/24 이 옵션은 패킷이 관련된 conntrack 항목의 원래 출발지 IP(original source IP)를 기반으로 일치시킨다. CIDR 형식의 넷마스크를 사용할 수도 있습니다.
--ctorigdst	iptables -A INPUT -p tcp -m conntrack --ctorigdst 192.168.0.0/24 이 옵션은 패킷의 목적지 IP를 기반으로 일치한다는 점을 제외하면 --ctorigsrc 옵션과 문법이 동일하다.
--ctreplsrc	iptables -A INPUT -p tcp -m conntrack --ctreplsrc 192.168.0.0/24 이 옵션은 패킷의 오리지널 응답 소스 IP(original conntrack reply source)를 기반으로 일치시키는데 사용한다. 기본적으로 --ctorigsrc와 동일하지만 대신 다가오는 패킷의 예상되는 응답 소스를 일치
--ctrepldst	iptables -A INPUT -p tcp -m conntrack --ctrepldst 192.168.0.0/24 이 옵션은 conntrack 항목의 응답 목적지 IP를 일치시킨다는 점을 제외하면 --ctreplsrc와 문법은 같다.

IP range Matches

옵션	예제 / 설명
--src-range	iptables -A INPUT -p tcp -m iprange --src-range 192.168.1.13-192.168.2.19 (-m iprange 를 반드시 지정해야 한다) source IP의 범위로 패킷을 일치시킨다
--dst-range	iptables -A INPUT -p tcp -m iprange --dst-range 192.168.1.13-192.168.2.19 destination IP의 범위로 패킷을 일치시킨다

State Matches

상태 일치는 커널의 연결 추적 코드(connection tracking code)와 함께 사용된다.
이를 통해 연결 상태를 알 수 있으며 ICMP 및 UDP와 같은 상태 비저장 프로토콜을 포함하여 거의 모든 프로토콜에서 동작한다.
모든 경우(All cases)에 연결에 대한 기본 타임아웃이 존재하고, 시간 초과되면 연결 추적 데이터베이스에서 삭제된다.
Rule에 -m state 문을 추가해 해당 일치 항목을 명시적으로 로드해야 한다.

옵션	예제 / 설명
--state	iptables -A INPUT -m state --state RELATED,ESTABLISHED 이 옵션은 일치시킬 패킷의 상태를 지정한다. 지정 가능한 상태는 다음 4가지 이다. 1. INVALID 2. ESTABLISHED 3. NEW 4. RELATED

TCP 연결 상태에 대한 간단한 설명

TCP 연결 상태 및 설명

상태	설명
커밍순...	