1. **Freebsd 설치**

가상머신 : Virtual Box 5.2.4

운영체제 : freebsd-11.1-RELEASE-i386-disc1.iso

(운영체제 설치 시 distribution select 에서 모두 체크)

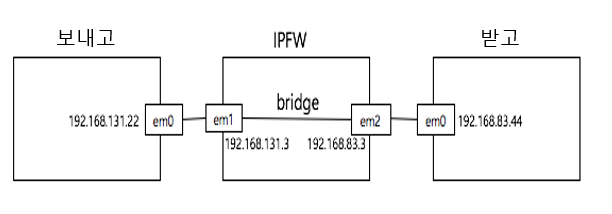
x window 설치 : <http://bubobubo.tistory.com/27> 참고

IPFW 활성화 : Freebsd\_ipfw setting.pdf 참고

설치에 관한 자세한 명령어는 Freebsd\_pkg install.hwp 참고

(현재 설치되어 있는 freebsd 비밀번호 : 0813)

1. **실습환경의 구성**



가상머신3개를 구축

네트워크 설정 및 bridge연결은 Freebsd\_bridge.hwp참고

Ipfw에 rule 추가 : add deny tcp from 192.168.131.22 to 192.168.83.44

(192.168.131.22 에서 192.168.83.44 로 보내는 tcp를 차단하겠다.)

tcpdump 또는 wireshark로 패킷 도착 확인

* 1. **Wireshark 확인**

# Pkg install wireshark : wireshark 설치 (180223 기준 2.4.3 버전)

**사용전에 입력하고 wireshark 켜야 한다.**

# chgrp network /dev/bpf\*

# chmod g+r /dev/bpf\*

Wireshark 접속 -> 인터페이스선택 -> 패킷확인

* 1. **TCP dump 확인**

# tcpdump -i em0 ( tcpdump로 em0에 들어오는 패킷을 확인하겠다. )

자세한 명령어는 <http://wnstjqdl.tistory.com/entry/%EB%A6%AC%EB%88%85%EC%8A%A4-tcpdump-%EC%82%AC%EC%9A%A9-%EB%B0%A9%EB%B2%95> 참고

1. **IPFW**

사용자 모드 :

/etc/rc.d/ipfw : 실행파일 동작을 위한 스크립트 문서

/sbin/ipfw : 실제 ipfw 실행파일

/usr/src/sbin/ipfw 디렉토리내에서 소스코드 수정 및 make로 실행파일 생성

이때 생성된 실행파일을 /sbin 으로 복사하고 reboot하면 적용이 된다.

커널 모드 :

/boot/kernel/ipfw.ko : ipfw.ko 실행파일로 동작

/usr/src/sys/netpfil/ipfw 디렉토리내에서 소스코드 수정

/usr/src/sys/modules/ipfw 디렉토리 내에서 make로 실행파일 생성

이때 생성된 실행파일을 /boot/kernel으로 복사하고 커널 전체 재 컴파일 하면된다.

(여기서 커널 전체 재 컴파일 명령어는 Freebsd\_ipfw setting.pdf 에 나와있다.)

* 1. **IPFW Rule**

Ipfw는 룰을 읽어서 동작이 된다. 룰에 관한 규칙 및 기능은 <https://www.freebsd.org/cgi/man.cgi?query=ipfw&apropos=0&sektion=0&manpath=FreeBSD+11.1-RELEASE+and+Ports&arch=default&format=html> 참고.

**등록된 룰을 읽어 드리는 부분** : /etc/rc.d 디렉토리 내 ipfw에서 rule을 읽어드린다.

실제 룰은 파일을 생성해서 룰을 등록을 해야한다. 그냥 add해서 추가된 룰은 재부팅 시 삭제가 된다. (자동으로 룰을 생성하는 기능을 추가할 예정)

* 1. **IPFW log**

Ipfw가 동작되면서 로그 기록을 남길 수 있다.

로그에 관한 명령어 및 기능은 <http://coffeenix.net/board_print.php?bd_code=695> 참고

(/var/log/ipfw/ipfw.log에 로그 기록이 남기도록 설정 해 놓았다.)

1. **Dummynet**

Dummynet은 트래픽 셰이퍼 및 네트워크 에뮬레이터이다.

/boot/kernel 디렉토리 내 dummynet.ko에 의해 작동

/usr/src/sys/netpfil/ipfw 디렉토리내에서 소스코드를 수정

/usr/src/sys/modules/dummynet 디렉토리내에서 make로 실행파일 생성

이때 생성된 실행파일을 /boot/kernel으로 복사하고 커널 전체 재 컴파일 하면된다.

(여기서 커널 전체 재 컴파일은 Freebsd\_ipfw setting.pdf 에 나와있다.)

더미넷 작동은Freebsd\_dummynet.docx 또는 dummynet.txt 참고

* 1. **Dummynet에 패킷이 들어오면 기록이 찍히는지 확인**

1. 가상 머신 1대로 확인 – 패킷이 들어오면 파일 입출력을 추가

* ping test 결과 : dummynet에 대한 기록은 남지 않는다.

1. 가상 머신 3대 ( 실습 환경 )로 확인 (핑 전송 후 방화벽 통과해 받는 환경)

* 네트워크 설정 및 브릿지 연결 : Freebsd\_bridge.hwp 참고
* ping test 결과 : dummynet에 대한 기록은 남지 않는다.

1. 방화벽에 웹 서버를 만들고 다른 가상 머신 에서 웹 서버 접속 시 기록 확인

* Ipfw에 웹 서버 설치 : apache24 설치 (180223기준 apache24가 최신버전)
* # pkg install apache24
* # vi /boot/loader.conf -> accf\_http\_load=”YES”
* # vi /etc/rc.conf -> apache24\_enable=”YES”
* # vi /usr/local/etc/apache24/httpd.conf -> ServerName 표시된 줄의 주석을 해제한 후 ServerName localhost:80 으로 수정
* # /usr/local/sbin/apachectl start : apache서버를 시작
* Test를 위해선 /usr/local/www/apache24/data 디렉토리 안에 html 을 추가하고 인터넷에 “ ip/filename”에 접속하면 된다.

(자세한 내용으로는 <http://inia30.tistory.com/7> 참고)

* test 결과 : dummynet에 대한 기록은 남지 않는다.

1. Printf( )문을 추가하여 ko파일 생성 및 커널 재 컴파일

* 테스트 결과 dummynet에 대한 기록은 남지 않는다.
* **Dummynet이 어느 부분에서 작동 되는건지 재 확인 필요**

1. **현재까지 진행 사항 정리**
2. **Ipfw.ko (커널모드)**

* 소스코드에 대한 정확한 분석은 아직 안함.
* Ipfw에 대한 로그 기록은 확인 -> ip\_fw\_log.c

1. **Dummynet.ko (커널모드)**

* ip\_dummynet.c : 시작점 예상
* int dummynet\_modevent() -> int dummynet\_io(): 패킷 i/o 진행 되는 함수 예상
* int dummynet\_io() -> dn\_ht\_find(), ipdn\_si\_find(), ipdn\_g\_find(), serve\_sched() 호출 : 패킷 플로우 세트 번호 찾기 및 스케줄러, i/o 진행

1. **ipfw (유저모드)**

* main.c : 시작점
* 크게 void help(), ipfw\_main(), ipfw\_readfile(), int main()으로 구성
* Ipfw2.c에서 add을 통한 룰 추가 및 컴파일, rule case별 구성 등 기능이 있다.

1. **디버깅**

* Freebsd\_gdb.hwp 참고
* 실행파일은 디버깅이 가능하나 ko파일에 대해서는 디버깅이 불가능

1. **Phsense (외국에서 판매되고있는 소형방화벽)**

* Phsense의 전체적 구조 및 설명은 phsense\_explain.pptx , pfsense\_structure.pptx 참고

1. **현재 운영체제가 openwrt로 변경**
2. **Freebsd 및 ipfw 는 사용 안한다. -> openwrt로 운영체제 확정 되었다.**
3. **Openwrt : c소스를 따로 볼 수 없다.**

* 이 부분에 대해서는 사용자 어플리케이션으로 접속해서 보도록 확인

(아직 가능한지는 미 확인)

1. **Ipfw.ko 파일 log기록 확인**

* /usr/src/sys/netpfil/ipfw/ip\_fw\_log.c 부분에 tcp가 printf()로 호출되는 부분이 있다. 그 부분에 test란 단어를 넣어서 tcp : 대신 tcp test : 로 호출되도록 수정을 한 후 make하여 ko파일을 /boot/kernel로 옮기고 커널 재 컴파일을 하여서 확인을 하였다.
* 이 부분에 대해서는 처음에 시도했을때는 /boot/kernel에 옮기고 reboot를 해도 시행이 되는 것을 확인했으나 회사 쪽 에서 알려준 결과 /boot/kernel에 옮긴 후 커널 재 컴파일을 해야 지 적용이 되는 것을 확인했다. 처음에 /boot/kernel에 옮기고 reboot 했을 때 작동 된 것은 어떠한 착오가 있는지 확인을 하지 못하였다.
* 결론으로, ko파일 적용을 위해서는 /boot/kernel로 옮긴 후 전체 커널 재 컴파일이 필요하다.
* 명령어로는 /usr/src/sys/netpfil/ipfw/ip\_fw\_log.c tcp 호출 부분에 test라는 단어 추가 -> /usr/src/sys/modules/ipfw 디렉토리에서 make -> ipfw.ko 실행파일 생성되면은 /boot/kernel에 복사 -> 커널 재 컴파일 (freebsd\_ipfw setting.pdf 에 나와있다.) -> /var/log/ipfw/ipfw.log에 기록이 남는지 확인

1. **Dummynet.ko 파일에 대한 기록 확인**

* 더미넷을 확인하기 위해 파일입출력 추가, 실습환경구축 후 확인, 웹서버 접속시 기록 확인, printf문을 추가하여 확인, 즉, 4가지 실습을 시행하였지만 어떠한 기록도 남지 않았다. (4.1번에 자세하게 정리 해 놓았다.)
* dummynet이 정상적으로 작동되는건지 파악이 불가능 하다.

1. **Ipfw 관련 참고 사이트**

Freebsd 홈페이지 : <https://www.freebsd.org>

Ipfw 메뉴얼 : <https://www.freebsd.org/doc/handbook/firewalls-ipfw.html>

Ipfw rule 관련 메뉴얼 :

<https://www.freebsd.org/cgi/man.cgi?query=ipfw&apropos=0&sektion=0&manpath=FreeBSD+11.1-RELEASE+and+Ports&arch=default&format=html>

Ipfw 메뉴얼 (한글) : <http://lifepatch.tistory.com/5?category=206026>

Ipfw log 메뉴얼 (한글) : <http://coffeenix.net/board_print.php?bd_code=695>

X window 구성 메뉴얼 : <http://bubobubo.tistory.com/27>

Dummynet 참고 : <http://info.iet.unipi.it/~luigi/dummynet/>

Freebsd 소스코드 : <https://github.com/freebsd/freebsd>

Openwrt 소스코드 : <https://github.com/openwrt/openwrt>

Pfsence 소스코드 : <https://github.com/pfsense/pfsense>

Ipfw 진행사항 정리 소스코드 : <https://github.com/ddytjq/ipfw>

Tcpdump 참고 : <http://wnstjqdl.tistory.com/entry/%EB%A6%AC%EB%88%85%EC%8A%A4-tcpdump-%EC%82%AC%EC%9A%A9-%EB%B0%A9%EB%B2%95>

Apache 참고 : <http://inia30.tistory.com/7>

리눅스 커널 모드 및 사용자 모드 에 대한 정보 : <https://kjvvv.kr/1766>