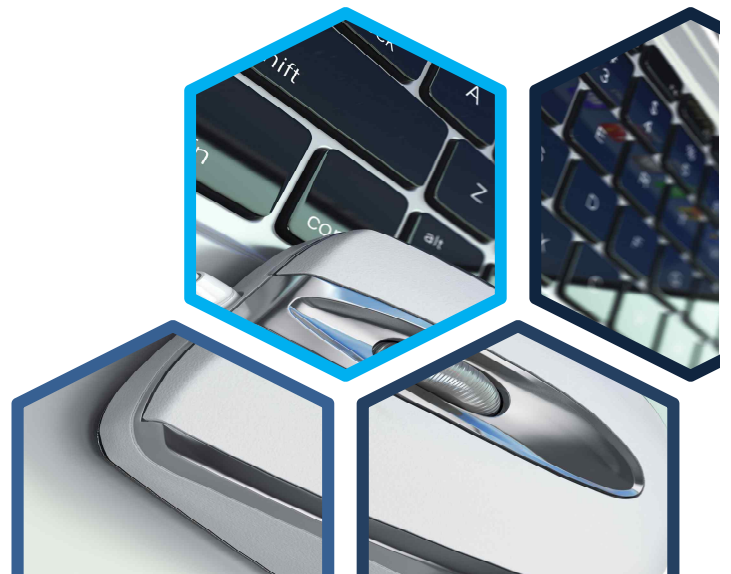




호스트 설정 및 네트워크 상태확인





학습목표

- 📊 호스트 이름 확인 방법을 설명할 수 있다.
- 📊 네트워크 상태 확인 방법을 설명할 수 있다.



학습내용

- 📊 호스트 이름 확인
- 📊 네트워크 상태 확인



호스트 이름 확인

1 관련 명령

uname -n
명령

hostname
명령

hostnamectl
명령



uname -n 명령

- 기능: 시스템 정보를 출력함
- 형식: uname [옵션]
- 옵션

-m	하드웨어 종류를 출력함
-n	호스트 이름을 출력함
-r	운영체제의 릴리스 정보를 출력함
-s	운영체제 이름을 출력함
-v	운영체제 버전을 출력함
-a	위의 모든 정보를 출력함

- 사용 예

```
uname -n  
uname -a
```



호스트 이름 확인

1 관련 명령



hostname 명령

- 기능: 호스트 이름을 출력하거나 설정함
- 형식: hostname [호스트 이름]
- 사용 예

```
hostname
hostname mail.han.server
```



hostnamectl 명령

- 기능: 시스템의 호스트 이름을 검색하거나 설정함
- 형식: hostnamectl [옵션] [명령]
- 옵션

-h	도움말을 출력함
--version	버전을 출력함

- 명령

status	현재 호스트 이름과 관련 정보를 출력함
set-hostname 호스트 이름	호스트 이름을 설정함

- 사용 예

```
hostnamectl
hostnamectl status
hostnamectl set-hostname mail.han.server
```



호스트 이름 확인

1 관련 명령



hostname 명령과 hostnamectl 명령

hostname 명령으로 호스트 이름 설정하기

호스트 이름 출력 및 설정하는
명령으로 `nmcli gen host`
명령으로도
같은 결과를 얻게 됨

hostnamectl 명령으로 호스트 이름 설정하기

시스템의 호스트 이름을 검색,
관리함



호스트 이름 확인

2 호스트 이름 저장



호스트 이름을 파일에 저장하기

- 페도라에서 호스트 이름을 저장하는 파일: `/etc/hostname`
- `/etc/hostname` 파일의 내용

```
[root@localhost ~]# cat /etc/hostname
localhost.localdomain
[root@localhost ~]#
```

- 도메인 이름을 포함한 호스트 이름만 저장
- 내용 수정 후 재시작 시 호스트 이름이 유지됨
- 호스트 이름을 새로 정의할 때는 한 네트워크에서 같은 이름을 사용하는 다른 호스트가 있으면 안 됨



호스트 이름 확인 실습 영상은
학습 콘텐츠에서 확인하실 수 있습니다.



네트워크 상태 확인

1 ping



개요

- IP 수준에서 **연결성을 확인**할 수 있는 도구
- ICMP 에코 요청 메시지를 전송
- 사용할 패킷 크기, 보낼 패킷 수, 사용된 경로 기록 여부, 사용할 TTL 값, Don't fragment 플래그 설정 여부를 지정 가능
- IPSec를 사용하는 컴퓨터의 경우
 - ping에 응답하기 전에 보안 연결을 구축하기 때문에 수 초의 시간 필요함



옵션

ping -n 10	ping 10개까지 전송
ping -l	ping 패킷의 Size를 정할 수 있음
ping -t	사용자가 취소할 때까지 실행
ping -f	Don't Fragment 플래그 설정으로 목적지까지 패킷이 도달하려면 네트워크 상황에 따라 패킷이 쪼개질 수 있는데 이 옵션을 사용하면 패킷이 쪼개지지 않음



네트워크 상태 확인

1 ping



통신 확인하기: ping

- 기능: 네트워크 장비에 **신호(ECHO_REQUEST)**를 보냄
- 형식: ping [옵션] [목적지 주소]
- 옵션

-a	통신이 되면 소리를 냄
-q	테스트 결과를 지속적으로 보여주지 않고 종합 결과만 출력함
-c 개수	보낼 패킷 수를 지정함

- 사용 예

```
ping 192.168.1.1  
ping -a www.e-koreatech.ac.kr
```

- 옵션 없이 사용하는 경우

- 패킷은 기본적으로 **56바이트의 크기 + 8바이트 헤더** 정보를 갖게 됨

-q 옵션 사용하기

아무 메시지도 출력되지 않다가
ctrl+C로 종료 시 통계 정보만
출력

-c 옵션 사용하기

보낼 패킷 수를 지정할 수 있음



네트워크 상태 확인

1 ping



도메인 이름을 사용하는 경우

- 보통 해당 도메인이 동작하는지 확인하기 위해 사용함
- 시스템에 따라서 보안을 강화하기 위해 ping 패킷이 왔을 때 응답하지 않도록 설정하는 경우도 있음
 - ping으로 연결되지 않는다고 해서 무조건 해당 시스템이 동작하지 않는다는 것은 아님



네트워크 상태 확인

2 traceroute(윈도우에서는 tracert이라 함)



개요

- IP 패킷이 컴퓨터에서 목적지로 가는데 사용한 모든 홉(패킷이 라우터에서 다른 라우터로의 이동하는 여정)을 보고자 할 때 이용
- 목적지까지 찾아가는 경로를 파악할 수 있으나, 귀환경로를 알려주는 것은 아님

Windows 95/98/NT/2000/XP



tracert

Linux



traceroute



reverse DN 점검 결과를 출력

- tracert 164.124.116.4 // Windows 계열
- traceroute 164.124.116.4 // UNIX 계열



reverse DN 점검 결과를 출력하지 않음

- 빠른 속도를 요하는 경우에 사용
- tracert -d 164.124.116.4 // Windows 계열
- traceroute -n 164.124.116.4 // UNIX 계열



네트워크 상태 확인

2 traceroute(윈도우에서는 tracert이라 함)



통신 경로 확인하기

- 기능: 목적지까지 **패킷이 거치는 경로**를 출력함
- 형식: traceroute [목적지 주소]
- 사용 예

```
traceroute 192.168.1.1  
traceroute www.e-koreatech.ac.kr
```



중간 노드의 기관 확인: **whois** 명령 사용

- 중간에 거치는 IP 주소가 어느 기관 것인지 알고 싶으면 **whois** 명령을 사용함



whois 명령이 설치되어 있지 않는 경우

- `sudo apt install whois` 명령으로 설치함



정상으로 경로가 확인되지 않는 경우

- 네트워크가 **연결되지 않는 구간**은 *가 출력
- 통신에 장애가 있거나 traceroute 명령을 거부한다는 의미임



네트워크 상태 확인

3 netstat



네트워크 상태 정보 출력하기: **netstat**

- 기능: 네트워크의 상태 정보를 출력함
- 형식: netstat [옵션]
- 옵션

-a	모든 소켓 정보를 출력함
-r	라우팅 정보를 출력함
-n	호스트명 대신에 IP 주소를 출력함
-i	모든 네트워크 인터페이스 정보를 출력함



쿼터 정보 확인하기: **netstat**

- 옵션

-s	프로토콜별로 네트워크 통계 정보를 출력함
-p	해당 소켓과 관련된 프로세스의 이름과 PID를 출력함

- 사용 예

```
netstat -rn  
netstat -s
```



네트워크 상태 확인

3 netstat



라우팅 테이블 확인하기: -r 옵션

- -r 옵션 사용시 라우팅 테이블을 확인할 수 있음
- -n 옵션을 같이 사용시 이름대신 IP 주소 출력



현재 열려 있는 포트 확인하기

- 현재 통신이 진행 중인 서비스는 해당 서비스 포트가 LISTEN 상태이므로 이를 통해 어떤 포트가 열려 있고 서비스 중인지 알 수 있음



현재 열려 있는 포트를 사용 중인 프로세스 확인하기: -p 옵션



인터페이스별 네트워크 통계 정보 확인하기: -i 옵션

RX-OK, TX-OK

정상적으로 주고받은 패킷의 개수

RX-ERR, RX-DROP, RX-OVR,
TX-ERR, TX-DROP, TX-OVR

송수신 중에 오류가 발생한 패킷의
개수



프로토콜별 네트워크 통계정보 확인: -s 옵션



네트워크 상태 확인

4 arp



MAC 주소와 IP 주소 확인하기: arp 명령

- 기능: arp 캐시 정보를 관리함
- 형식: arp [IP 주소]
- 사용 예

```
arp  
arp 192.168.1.1
```

- 옵션 없이 사용할 경우: 현재 같은 네트워크에 연결되어 있는 시스템의 MAC 주소와 IP 주소를 출력
- 특정 시스템의 MAC 주소를 확인



네트워크 상태 확인

5 arp



패킷 캡처하기: tcpdump

- 기능: 네트워크상의 **트래픽을 덤프함**
- 형식: tcpdump [옵션]
- 옵션

-c 패킷 수	지정한 패킷 수만을 덤프 받고 종료함
-i 인터페이스명	특정 인터페이스를 지정함
-n	IP 주소를 호스트명으로 바꾸지 않음
-q	정보를 간단한 형태로 보여줌
-x	패킷의 내용을 16진수와 ASCII로 출력함
-w 파일명	덤프한 내용을 지정한 파일에 저장함
-r 파일명	덤프를 저장한 파일에서 읽어옴
host 호스트명 또는 주소	지정한 호스트가 받거나 보낸 패킷만 덤프함
tcp port 번호	지정한 포트 번호 패킷만 덤프함
ip	IP 패킷만 덤프

사용 예

```
tcpdump
tcpdump -i eth0
tcpdump -i eth0 -w DUMP.out
tcpdump tcp port 22 and host 192.168.0.7
```



네트워크 상태 확인

5 arp



옵션 없이 사용하는 경우

- ▶ 현재 시스템에서 주고받는 모든 패킷을 캡처하여 **패킷의 헤더 부분 정보**를 출력



캡처할 패킷 개수 지정하기: **-c 옵션**

예 ▶ 패킷을 세 개만 캡처

```
[root@localhost ~]# tcpdump -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
```



캡처한 패킷 정보를 파일로 저장하기: **-w 옵션**

예 ▶ 패킷 세 개를 캡처하여 **dump.out** 파일에 저장



특정 포트로 송수신되는 패킷 캡처하기: **tcp port 옵션**

예 ▶ 192.168.0.13과 주고받는 패킷 중에서 **22번 포트에** 해당하는 패킷 세 개를 캡처

```
[root@localhost ~]# tcpdump -c 3 tcp port 22 and host 192.168.1.13
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
```




네트워크 상태 확인

5 arp



캡처한 내용을 ASCII로 보기: **-X 옵션**

```
[root@localhost ~]# tcpdump -Xqr dump.out  
reading from file dump.out, link-type EN10MB (Ethernet)
```






네트워크 상태 확인 실습 영상은
학습 콘텐츠에서 확인하실 수 있습니다.








핵심요약

1 호스트 이름 확인

-  `uname -n` 명령: 시스템 정보를 출력
-  `hostname` 명령: 호스트 이름을 출력하거나 설정함
-  `hostnamectl` 명령: 시스템의 호스트 이름을 검색하거나 설정함

2 네트워크 상태 확인

-  `ping`: IP 수준에서 연결성을 확인할 수 있는 도구임
-  `traceroute`: IP 패킷이 컴퓨터에서 목적지로 가는데 사용한 모든 홉을 보고자 할 때 이용함
-  `netstat`: 네트워크의 상태 정보를 출력함
-  `arp`: arp 캐시 정보를 관리함
-  `tcpdump`: 네트워크상의 트래픽을 덤프함