INNOVATION
PAI CHAI UNIVERSITY

도전학기제(꿈드림설계) 과제 수행 중간보고서

2024-2학기

성명	한 승 연	**학과(전공)**	소프트웨어공학부(정보보안학)
학번	2184014	**학년**	3
참여유형	전공심화 고도화형	**신청과목명**	꿈드림설계2
프로젝트 명	네트워크 보안 장비 제작		
주제	네트워크 보안 솔루션 연구 및 장비 제작		
책임교수	**소속**	소프트웨어공학부 정보보안학	
	성명	함 형 민	

	주차	**주요활동 내용**			
		최초 계획(계획서 참고)		**실제 활동내용 및 목표달성 여부**	**투입 시간**
프로젝트 수행사항	1주차	**팀 목표 및 활동**	네트워크 이해 1. 네트워크 패킷 통신 과정 2. 프로토콜 3. 3 Way Handshake & 4 Way Handshake 4. 네트워크 장비 기능	[네트워크 이해] 1. 네트워크 패킷 통신 과정(3 Way Handshake & 4 Way Handshake) 학습 2. 프로토콜 종류 학습	9
		개인 목표 및 활동	1. 네트워크 패킷 통신 과정 이해 2. 프로토콜 종류 학습 3 Way Handshake 및 4 Way Handshake 이해	1.TCP 3 Way HandShake 과정 정의: 클라이언트와 서버 간의 연결을 맺기 위한 절차이다. 1-1. 클라이언트는 서버에 접속을 요청하는 SYN(M) 패킷을 보냄. - SYN 플래그: 1로 설정되어 있음을 나타낸다. - 시퀀스 번호(Sequence Number): 클라이언트가 생성한 초기 시퀀스 번호(ISN: Initial Sequence Number)를 포함한다. - 이 번호는 세션 동안 각 바이트에 대한 고유 번호를 부여하는 데 사용됩니다. 해당 패킷은 ACK 플래그가 0으로 설정되어 있음을 나타낸다. 1-2. 서버는 클라이언트의 요청 패킷을 받고 요청을 수락한다는 ACK(M+1)과 SYN(N)이 설정된 패킷을 발송한다. - SYN 플래그: 1로 설정되어 있음을 나타낸다.	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			- ACK 플래그: 1로 설정되어 있으며, 클라이언트의 SYN 패킷을 확인했음을 나타낸다. - Sequence Number: 서버가 생성한 초기 시퀀스 번호를 포함한다. - Acknowledgment Number: 클라이언트의 시퀀스 번호에 1을 더한 값입니다. 이는 서버가 클라이언트의 다음 데이터를 기대하는 시퀀스 번호를 나타낸다. 1-3. 클라이언트는 서버의 수락 응답 패킷을 받고 ACK(N+1)을 서버로 보낸다. 연결이 성립된다. - ACK 플래그: 1로 설정되어 있으며, 서버의 SYN-ACK 패킷을 확인했음을 나타낸. - 시퀀스 번호(Sequence Number): 클라이언트의 초기 시퀀스 번호에 1을 더한 값입니다. - 인정 번호(Acknowledgment Number): 서버의 시퀀스 번호에 1을 더한 값이다. 이는 클라이언트가 서버의 다음 데이터를 기대하는 시퀀스 번호를 나타낸다. 2. TCP 4 WAY-HandShake 정의: 3-way-HandShake는 TCP의 연결을 초기화할 때 사용한다면, 4-way-HandShake는 세션을 종료하기 위해 수행되는 절차이다. 2-1. 클라이언트가 연결을 종료하겠다는 FIN 플래그를 전송한다. 이때 클라이언트는 FIN_WAIT 상태가 된다. 2-2. 서버는 클라이언트의 요청을 받고, 알겠다는 확인 메세지로 ACK를 보낸다. 그리고 나서 데이터를 모두 보낼 때까지 잠깐 TIME_OUT이 된다. 이때 서버는 CLOSE_WAIT 상태가 된다. 2-3. 데이터를 모두 보내고 통신이 끝났으면, 연결이 종료되었다는 FIN 플래그를 전송한다. 이때 서버는 LAST_ACK 상태가 된다. 클라이언트는 종료 메세지를 확인하였다는 ACK를 보낸다. 서버는 ACK 메세지를 받고 소켓 연결을 CLOSE한다. 클라이언트는 아	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			직 서버로부터 받지 못한 데이터가 있을 것을 대비해 일정 시간동안 세션을 남겨놓고 잉여 패킷을 기다린다(TIME_WAIT). 3. HTTP 통신(GET,POST,DELETE,PETCH 등) 3-1. HTTP 통신은 웹 서버와 클라이언트 간의 데이터 소통을 위해 사용되는 프로토콜이다. 다양한 HTTP 메서드를 통해 이루어지며, REST API 규칙에 따라 통신된다. REST API는 CRUD(Create, Read, Update, Delete) 작업으로 구성되어 있다. 각 HTTP 메서드는 다음과 같은 용도로 사용된다. 3-2. GET: 웹페이지에서 데이터를 조회할 때 사용된다. 주로 서버에 있는 자원의 상태를 요청한다. 3-3. POST: 서버에 새로운 데이터를 생성하거나 제출할 때 사용된다. 대량의 데이터를 전송할 수 있다. 3-4. PATCH: 기존의 데이터를 부분적으로 수정할 때 사용된다. 전체 데이터를 교체하는 것이 아니라, 필요한 부분만 업데이트한다. 3-5. DELETE: 서버에서 특정 데이터를 삭제할 때 사용된다. 4. 네트워크 패킷 통신 과정 1. 패킷 분할과 라우팅 - 송신자 컴퓨터에서 데이터를 패킷으로 분할한다. 각 패킷은 고유한 ID와 목적지 주소를 가지며, 패킷 헤더에 저장된다. - 이후 송신자 컴퓨터에서는 패킷을 다음 라우터로 보내기 위해 라우팅 결정을 한다. 이때는 다양한 라우팅 알고리즘이 사용됨. 2. 라우터에서 패킷 처리 - 라우터는 패킷의 목적지 주소를 확인하여 다음 라우터로 패킷을 전송한다. 이때 라우터는 패킷 헤더에 저장된 목적지 주소를 기반으로 패킷을 전송한다.	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			- 라우터는 패킷을 처리하면서 다양한 기술과 프로토콜을 사용한다. 예를 들어, 라우터는 패킷을 재전송하거나, 패킷의 우선순위를 조정할 수 있다. 3. 패킷 전송 및 중계 - 패킷은 다음 라우터로 전송된다. 이때는 라우터 간의 연결이 필요하며, 이를 위해 다양한 기술과 프로토콜이 사용된다. - 패킷이 전송되는 동안에는 중간에 여러 문제가 발생할 수 있다. 예를 들어, 패킷이 분실되거나 손상될 수 있다. 이러한 문제를 처리하기 위해 다양한 오류 검출과 복구 기술이 사용됨. 4. 패킷 수신과 재조립 - 패킷이 목적지에 도달하면, 수신자 컴퓨터에서 패킷을 수신한다. 이대는 패킷의 ID와 목적지 주소를 확인하여 정확한 패킷을 수신한다. - 이후 패킷은 재조립되어 원래의 데이터로 복원된다. 이 데이터는 소프트웨어나 하드웨어로 처리되어 최종적으로 사용자에게 제공된다. 5. 프로토콜 종류(HTTP,HTTPS,TCP,UDP) 1. HTTP: 월드 와이드 웹의 토대이며 하이퍼텍스트 링크를 사용하여 웹 페이지를 로드하는 데 사용됨. HTTP는 네트워크 장치 간에 정보를 전송하도록 설계된 애플리케이션 계층 프로토콜이며 네트워크 프로토콜 스택의 다른 계층 위에서 실행된다. 2. HTTPS: 웹 브라우저와 웹 사이트 간에 데이터를 전송하는데 사용되는 기본 프로토콜인 HTTP의 보안 버전이다. 3. TCP: 두 개의 호스트를 연결하고 데이터 스트림을 교환하게 해주는 네트워크 프로토콜이다. 4. UDP: 통신 프로토콜이며, 특히 비디오 재생 또는 DNS 조회와 같이 시간에 민감한 전송을 위해 인터넷을 제공한다. 이 프로토콜을 사용하면 데이터가 전송되기 전에는 공식적으로 연결이 설정되지 않으므로 통신 속	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			도가 빨라진다. 2-3. 데이터를 모두 보내고 통신이 끝났으면, 연결이 종료되었다는 FIN 플래그를 전송한다. 이때 서버는 LAST_ACK 상태가 된다. 클라이언트는 종료 메시지를 확인하였다는 ACK를 보낸다. 서버는 ACK 메세지를 받고 소켓 연결을 CLOSE한다. 클라이언트는 아직 서버로부터 받지 못한 데이터가 있을 것을 대비해 일정 시간동안 세션을 남겨놓고 잉여 패킷을 기다린다 (TIME_WAIT). 3. HTTP 통신(GET,POST,DELETE,PETCH 등) 3-1. HTTP 통신은 웹 서버와 클라이언트 간의 데이터 소통을 위해 사용되는 프로토콜이다. 다양한 HTTP 메서드를 통해 이루어지며, REST API 규칙에 따라 통신된다. REST API는 CRUD(Create, Read, Update, Delete) 작업으로 구성되어 있다. 각 HTTP 메서드는 다음과 같은 용도로 사용된다. 3-2. GET: 웹페이지에서 데이터를 조회할 때 사용된다. 주로 서버에 있는 자원의 상태를 요청한다. 3-3. POST: 서버에 새로운 데이터를 생성하거나 제출할 때 사용된다. 대량의 데이터를 전송할 수 있다. 3-4. PATCH: 기존의 데이터를 부분적으로 수정할 때 사용된다. 전체 데이터를 교체하는 것이 아니라, 필요한 부분만 업데이트한다. 3-5. DELETE: 서버에서 특정 데이터를 삭제할 때 사용된다. 4. 네트워크 패킷 통신 과정 1. 패킷 분할과 라우팅	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			- 송신자 컴퓨터에서 데이터를 패킷으로 분할한다. 각 패킷은 고유한 ID와 목적지 주소를 가지며, 패킷 헤더에 저장된다. - 이후 송신자 컴퓨터에서는 패킷을 다음 라우터로 보내기 위해 라우팅 결정을 한다. 이때는 다양한 라우팅 알고리즘이 사용됨. 2. 라우터에서 패킷 처리 - 라우터는 패킷의 목적지 주소를 확인하여 다음 라우터로 패킷을 전송한다. 이때 라우터는 패킷 헤더에 저장된 목적지 주소를 기반으로 패킷을 전송한다. - 라우터는 패킷을 처리하면서 다양한 기술과 프로토콜을 사용한다. 예를 들어, 라우터는 패킷을 재전송하거나, 패킷의 우선순위를 조정할 수 있다. 3. 패킷 전송 및 중계 - 패킷은 다음 라우터로 전송된다. 이때는 라우터 간의 연결이 필요하며, 이를 위해 다양한 기술과 프로토콜이 사용된다. - 패킷이 전송되는 동안에는 중간에 여러 문제가 발생할 수 있다. 예를 들어, 패킷이 분실되거나 손상될 수 있다. 이러한 문제를 처리하기 위해 다양한 오류 검출과 복구 기술이 사용됨. 4. 패킷 수신과 재조립 - 패킷이 목적지에 도달하면, 수신자 컴퓨터에서 패킷을 수신한다. 이때는 패킷의 ID와 목적지 주소를 확인하여 정확한 패킷을 수신한다. - 이후 패킷은 재조립되어 원래의 데이터로 복원된다. 이 데이터는 소프트웨어나 하드웨어로 처리되어 최종적으로 사용자에게 제공된다. 5. 프로토콜 종류(HTTP,HTTPS,TCP,UDP) 1. HTTP: 월드 와이드 웹의 토대이며 하이퍼텍스트 링크를 사용하여 웹 페이지를 로드하는 데 사용됨. HTTP는 네트워크 장치 간에 정보를 전송하도록 설계된	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			애플리케이션 계층 프로토콜이며 네트워크 프로토콜 스택의 다른 계층 위에서 실행된다. 2. HTTPS: 웹 브라우저와 웹 사이트 간에 데이터를 전송하는데 사용되는 기본 프로토콜인 HTTP의 보안 버전이다. 3. TCP: 두 개의 호스트를 연결하고 데이터 스트림을 교환하게 해주는 네트워크 프로토콜이다. 4. UDP: 통신 프로토콜이며, 특히 비디오 재생 또는 DNS 조회와 같이 시간에 민감한 전송을 위해 인터넷을 제공한다. 이 프로토콜을 사용하면 데이터가 전송되기 전에는 공식적으로 연결이 설정되지 않으므로 통신 속도가 빨라진다.	
2주차	**팀 목표 및 활동**	내 • 외부 네트워크 구축 1. 유 • 무선 라우터를 활용한 내 • 외부 네트워크 구축 2. 라우터 기능 분석 및 보완(발전) 사항 파악	[내 • 외부 네트워크 구축] - 유 • 무선 라우터를 활용한 내 • 외부 네트워크 구축 - 라우터 기능 분석 및 보완(발전) 사항 파악	9
	개인 목표 및 활동	1. 내 • 외부 네트워크 구현 2. 라우터 기능 분석 및 보완(발전) 사항 파악	유무선 라우터의 기능 DHCP 서버: DHCP(Dynamic Host Configuration Protocol) 서버는 네트워크에 접속하는 장치에 자동으로 IP 주소를 할당하는 기능이다. 이를 통해 사용자는 수동으로 IP 주소를 설정할 필요 없이, 라우터가 자동으로 IP 주소와 서브넷 마스크, 게이트웨이 주소 등을 제공받을 수 있다. 이 기능은 네트워크 관리의 효율성을 높이고, IP 주소 충돌을 방지하는 데 중요한 역할을 한다. NAT (Network Address Translation): NAT는 내부 네트워크의 사설 IP 주소와 외부 네트워크의 공인 IP 주소 간의 변환을 담당하는 기능이다. 이를 통해 여러 대의 장치가 하나의 공인 IP 주소를 공유하여 인터넷	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			에 접속할 수 있게 된다. NAT는 보안 측면에서도 유용하다. 외부에서 내부 네트워크의 장치에 직접 접근하는 것을 차단하여, 내부 네트워크의 구조를 외부에 노출시키지 않기 때문이다. 무선 AP (Access Point): 무선 AP는 무선 인터넷 연결을 제공하는 장치이다. 유무선 라우터는 일반적으로 무선 AP의 기능을 내장하고 있어, 사용자는 Wi-Fi를 통해 무선으로 인터넷에 접속할 수 있다. 무선 AP는 기기 간의 무선 통신을 가능하게 하며, 네트워크의 범위를 확장하는 데 도움을 준다. 이를 통해 모바일 기기나 노트북 등 다양한 장치가 유선 연결 없이도 인터넷에 연결될 수 있다. 보완 및 발전 사항 본 실험 과정에서 유무선 라우터의 보안 기능에 대해 살펴본 결과, 몇 가지 발전이 필요한 사항을 발견하였다. 방화벽 기능: 유무선 라우터에 내장된 방화벽 기능은 외부에서 오는 공격을 차단하고 내부 네트워크를 보호하는 데 필수적이다. 그러나 일부 라우터는 기본적인 방화벽만을 제공하며, 고급 설정이나 사용자 정의가 부족할 수 있다. 따라서 보다 강력한 보안 체계를 구축하기 위해서는 방화벽 기능의 향상이 필요하다. 패킷 분석 및 로그 기능: 네트워크에서 발생하는 트래픽을 실시간으로 모니터링하고, 패킷을 분석하는 기능도 중요한 보안 요소이다. 이를 통해 비정상적인 트래픽 패턴이나 잠재적인 공격을 조기에 발견할 수 있다. 하지만 많은 유무선 라우터는 이러한 기능이 부족하거나 기본적인 로그만을 제공하여, 보다 심층적인 분석을 위한 추가적인 솔루션이 요구된다.	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			결론적으로, 유무선 라우터는 기본적인 네트워크 구축과 관리에 매우 유용하지만, 보안 측면에서의 기능이 강화되어야 보다 안전한 네트워크 환경을 제공할 수 있다. 이러한 발전 사항을 염두에 두고, 향후 네트워크 구축 시 보안 기능을 더욱 강화하는 방향으로 진행하는 것이 중요하다.	
3주차	팀 목표 및 활동	네트워크 장비 제작 1. dhcpcd 학습 2. hostapd 학습 3. dnsmasq 학습 4. 미니 PC에 네트워크 응용 소프트웨어 설치 후 무선 라우터(Wireless Router) 제작	[네트워크 장비 제작] - dhcpcd 학습 - hostapd 학습 - dnsmasq 학습 - 미니 PC에 네트워크 응용 소프트웨어 설치 후 무선 라우터(Wireless Router) 제작	9
	개인 목표 및 활동	서영석 1. 네트워크 응용 소프트웨어 dhcpcd, hostapd, dnsmasq 학습 2. 네트워크 장비 제작 한승연 1. 네트워크 응용 소프트웨어 탐색 및 사용법(명령어) 학습 • 공유	1. DHCP 클라이언트: dhcpcd dhcpcd는 리눅스에서 동적으로 IP 주소를 할당받기 위해 사용되는 DHCP 클라이언트이다. bash # dhcpcd 설치 sudo apt-get install dhcpcd # dhcpcd 서비스 시작 sudo systemctl start dhcpcd # 부팅 시 자동 시작 설정 sudo systemctl enable dhcpcd 2. 경량 DNS 및 DHCP 서버: dnsmasq dnsmasq는 소규모 네트워크에서 쉽게 설정할 수 있는 DNS 포워더 및 DHCP 서버이다. bash # dnsmasq 설치 sudo apt-get install dnsmasq # dnsmasq 서비스 시작 sudo systemctl start dnsmasq	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			# 부팅 시 자동 시작 설정 sudo systemctl enable dnsmasq # 설정 파일 수정 (예: /etc/dnsmasq.conf) # sudo nano /etc/dnsmasq.conf 3. 무선 액세스 포인트 관리: hostapd hostapd는 무선 액세스 포인트(AP)를 설정하고 관리하는 데 사용되는 서비스이다. bash # hostapd 설치 sudo apt-get install hostapd # hostapd 서비스 시작 sudo systemctl start hostapd # 부팅 시 자동 시작 설정 sudo systemctl enable hostapd # 설정 파일 수정 (예: /etc/hostapd/hostapd.conf) # sudo nano /etc/hostapd/hostapd.conf 4. 네트워크 트래픽 제어: iptables iptables는 네트워크 트래픽을 제어하는 방화벽이다. bash # iptables 설치 sudo apt-get install iptables # 기본 정책 설정 sudo iptables -P INPUT ACCEPT sudo iptables -P FORWARD ACCEPT sudo iptables -P OUTPUT ACCEPT # 특정 포트 열기 (예: 80 포트) sudo iptables -A INPUT -p tcp --dport	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			80 -j ACCEPT # 규칙 저장 sudo iptables-save > /etc/iptables/rules.v4 네트워크 장비 제작 과정 요약 미니 PC에 리눅스 운영체제 설치. dnsmasq, hostapd, iptables 설치. dnsmasq 및 hostapd 설정 파일 수정. iptables 규칙 설정.	
4주차	팀 목표 및 활동	네트워크 패킷 캡처 및 분석 1. Wireshark 학습 및 실습 2. 네트워크 패킷 캡처 및 분석	[네트워크 패킷 캡처 및 분석] - Wireshark 학습 및 실습 - 네트워크 패킷 캡처 및 분석	9
	개인 목표 및 활동	서영석 1. Wireshark 학습 및 실습 2. 네트워크 패킷 캡처 및 분석 한승연 1. 네트워크 패킷 캡처 및 분석 소프트웨어 탐색 및 사용법(명령어) 학습 • 공유	1. 인터페이스 Wireshark GUI(그래픽 사용자 인터페이스) 제공 실시간으로 패킷 목록 확인 선택한 패킷의 세부 정보 시각적으로 분석 가능 소스 및 목적지 MAC 주소, IP 주소, 포트 번호 등의 정보 제공 tcpdump CLI(명령줄 인터페이스) 도구 명령어 입력으로 패킷 캡처 및 분석 실시간 출력 또는 파일 저장 가능 스크립트를 통한 자동화 가능 2. 사용 용도 Wireshark 네트워크 트래픽 상세 분석 이메일, 파일 전송 내용 등 데이터 확인 문제 발생 시 원인 파악에 유용 tcpdump 빠르고 간단한 패킷 캡처 특정 트래픽 필터링 가능 초기 캡처 및 반복 작업 자동화에 적합 3. 데이터 저장 및 분석	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			Wireshark 실시간 패킷 캡처 및 파일 저장 패킷 필터링 및 통계 기능 활용 IO 그래프 및 프로토콜 계층 뷰 제공 tcpdump 명령어로 패킷 파일 저장 Wireshark와 같은 도구에서 분석 가능 저장된 데이터 유연하게 관리 가능 4. 성능 관리 및 통계 Wireshark 다양한 성능 분석 기능 제공 IO 그래프 및 특정 프로토콜 분석 가능 트래픽 생성 노드 식별 tcpdump 성능 통계 기능 제한적 특정 조건에 따라 패킷 필터링 가능 빠른 패킷 캡처 속도 제공 5. 사용 사례 Wireshark 네트워크 트래픽 식별 및 성능 문제 해결 복잡한 분석에 적합 네트워크 성능 모니터링 가능 tcpdump 신속한 캡처 및 간단한 분석 CLI 환경에서 사용 후 Wireshark로 결과 분석 가능 6. Wireshark 명령어 정리 IP 주소 기반 필터링 출발지 혹은 도착지 IP 주소 검색 ip.addr == 192.168.0.1 출발지 혹은 도착지 IP가 192.168.0.1인 패킷 표시 도착지 IP 주소 검색 ip.dst == 192.168.0.1	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			도착지 IP가 192.168.0.1인 패킷 표시 출발지 IP 주소 검색 ip.src == 192.168.0.1 출발지 IP가 192.168.0.1인 패킷 표시 포트 기반 필터링 TCP 포트 5000 검색 tcp.port == 5000 TCP 포트 5000으로 전송되는 패킷 표시 UDP 포트 5001 검색 udp.port == 5001 UDP 포트 5001으로 전송되는 패킷 표시 논리적 연산자 조합 AND: 두 개 이상의 조건이 모두 참일 때 참 OR: 두 개 이상의 조건 중 하나라도 참이면 참 NOT: 조건이 거짓일 때 참 무선랜 MAC 주소 기반 필터링 출발지 또는 목적지 MAC 주소 검색 wlan.addr == 00:0C:29:78:96:2C 출발지 혹은 목적지 MAC 주소가 00:0C:29:78:96:2C인 패킷 표시 출발지 MAC 주소 검색 wlan.sa == 00:0C:29:78:96:2C 출발지 MAC 주소가 00:0C:29:78:96:2C인 패킷 표시 목적지 MAC 주소 검색 wlan.da == 00:0C:29:78:96:2C 목적지 MAC 주소가 00:0C:29:78:96:2C인	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			패킷 표시 이더넷 MAC 주소 기반 필터링 출발지 또는 목적지 MAC 주소 검색 eth.addr == 00:0C:29:1D:1F:0F 출발지 혹은 목적지 MAC 주소가 00:0C:29:1D:1F:0F인 패킷 표시 출발지 MAC 주소 검색 eth.src == 00:0C:29:1D:1F:0F 출발지 MAC 주소가 00:0C:29:1D:1F:0F인 패킷 표시 목적지 MAC 주소 검색 eth.dst == 00:0C:29:1D:1F:0F 목적지 MAC 주소가 00:0C:29:1D:1F:0F인 패킷 표시 기타 필터링 도착지가 해당 IP인 패킷 검색 ip.dst_host == 192.168.0.13 도착지가 192.168.0.13인 패킷 검색 도착지가 해당 IP이면서 프로토콜이 HTTP인 패킷 검색 ip.dst_host == 192.168.0.13 && http 도착지가 192.168.0.13이면서 프로토콜이 HTTP인 패킷 검색 프레임 길이가 128보다 작거나 같은 패킷 검색 frame.len <= 128 프레임 길이가 128바이트 이하인 패킷 검색	
5주차	팀 목표	네트워크 위협 분석 체계(NTAS,	[네트워크 위협 분석 체계(NTAS,	9

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
	및 활동	Network Traffic Analysis System) 제작 1. 네트워크 장비에 Wireshark 설치 후 가동 2. 네트워크 패킷 캡처 및 분석	Network Traffic Analysis System) 제작] - 네트워크 장비에 Wireshark 설치 후 가동 - 네트워크 패킷 캡처 및 분석	
	개인 목표 및 활동	서영석,한승연 1. 네트워크 장비에 Wireshark 설치 및 가동 2. 네트워크 패킷 캡처 및 분석	네트워크 패킷 캡처 및 분석 실험 NTAS의 성능을 검증하기 위해 네트워크 패킷 캡처 및 분석 실험을 진행하였다. 이 실험의 목적은 내부 네트워크의 PC가 외부 네트워크와 통신할 때, NTAS에서 패킷을 성공적으로 캡처하고 분석할 수 있는지를 확인하는 것이다. 실험 환경 구성 실험을 위해 다음과 같은 네트워크 구성을 설정하였다: PC(내부 네트워크): 사용자 디바이스로, 내부 네트워크에 연결되어 있으며 외부 네트워크와 통신을 시도한다. 무선 라우터(NTAS): 실험의 핵심 장비로, 내부 네트워크와 외부 네트워크 사이의 중계 역할을 수행하며, Wireshark를 통해 실시간 트래픽을 모니터링한다. 외부 네트워크: 인터넷을 포함한 외부 네트워크로, 내부 PC에서 발생하는 요청이 전달되고, 해당 트래픽이 라우터를 통해 외부로 나가는 구조이다. 이러한 구성으로, 내부 네트워크의 트래픽이 모두 무선 라우터를 통과하는 방식으로 설계되었다. 이를 통해 무선 라우터에서 캡처되는 모든 트래픽을 Wireshark로 분석할 수 있었다. 실험 과정 단계 1: 내부 네트워크 통신 먼저, 내부 네트워크에 연결된 PC에서 외부 네트워크로 데이터 요청을 전송하였다. 예를 들어, 웹 브라우저를 통해 외부 서버에 접속하거나 파일을 다운로드하는 등의 네트워크 활동을 수행하였다. 단계 2: 무선 라우터(NTAS)에서 패킷 캡처 PC가 외부 네트워크와 통신할 때, 모든 트래픽은 무선 라우터(NTAS)를 거쳐간다. 이	

주차	주요활동 내용			
	최초 계획(계획서 참고)		실제 활동내용 및 목표달성 여부	투입 시간
			때 NTAS에 설치된 Wireshark가 해당 트래픽을 실시간으로 모니터링하고, 이를 패킷 단위로 캡처하였다. 캡처된 패킷은 IP 주소, 프로토콜 정보, 포트 번호, 데이터 내용 등을 포함하고 있어, 네트워크 상태를 종합적으로 분석할 수 있다. 단계 3: 패킷 분석 및 위협 탐지 캡처된 패킷을 분석하여 트래픽의 종류를 확인하고, 비정상적인 패킷이 있는지를 탐지하는 과정을 진행하였다. Wireshark를 통해 캡처된 패킷을 시각적으로 분석하며, 특히 악의적인 트래픽이나 보안 취약점이 있는 패킷을 탐지하는 데 중점을 두었다. 이 과정에서 트래픽 유형별로 필터링 기능을 사용해, 특정 프로토콜이나 비정상적인 패킷을 선별적으로 분석할 수 있었다. 실험 결과 실험 결과, **무선 라우터(NTAS)**를 통해 통신되는 모든 트래픽을 성공적으로 캡처하고, Wireshark를 사용하여 세부적으로 분석할 수 있음을 확인하였다. 이를 통해 네트워크 내부에서 발생할 수 있는 잠재적인 보안 위협을 사전에 감지할 수 있었다. 또한, 특정 통신에서 의심스러운 트래픽이나 비정상적인 패킷 흐름이 발생할 경우, 이를 실시간으로 탐지하고 경고할 수 있는 체계가 구축되었음을 검증하였다.	

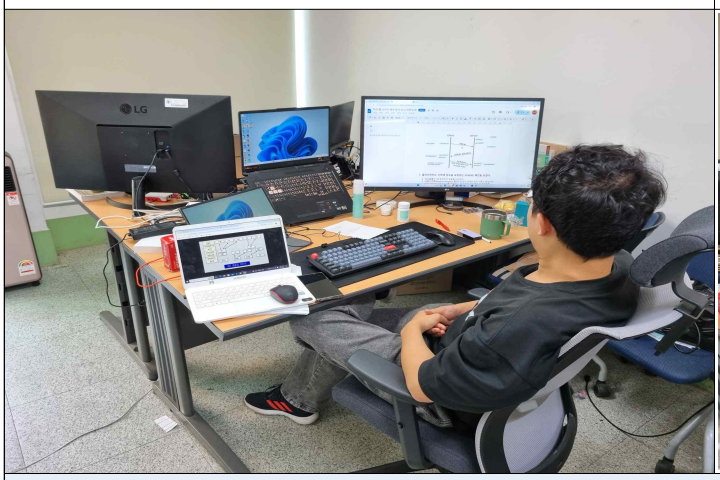
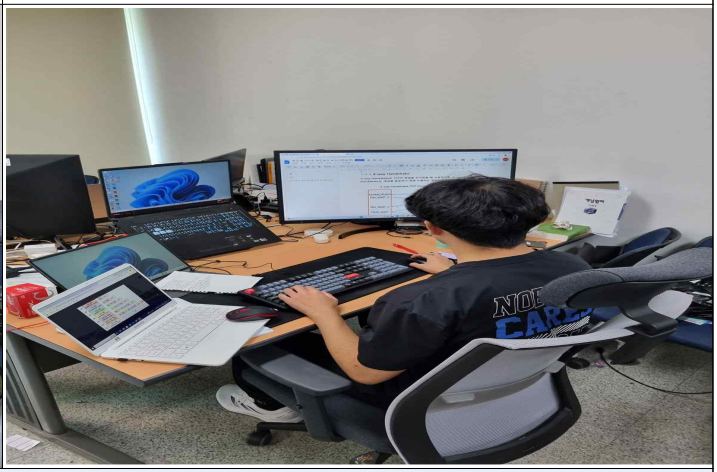
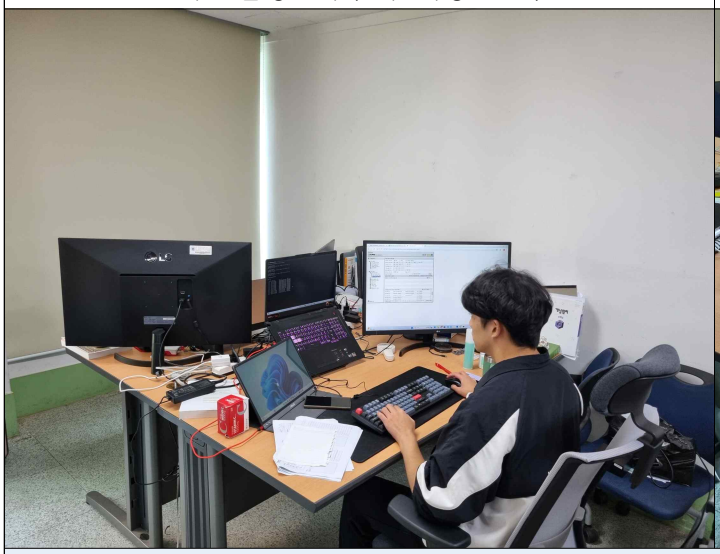
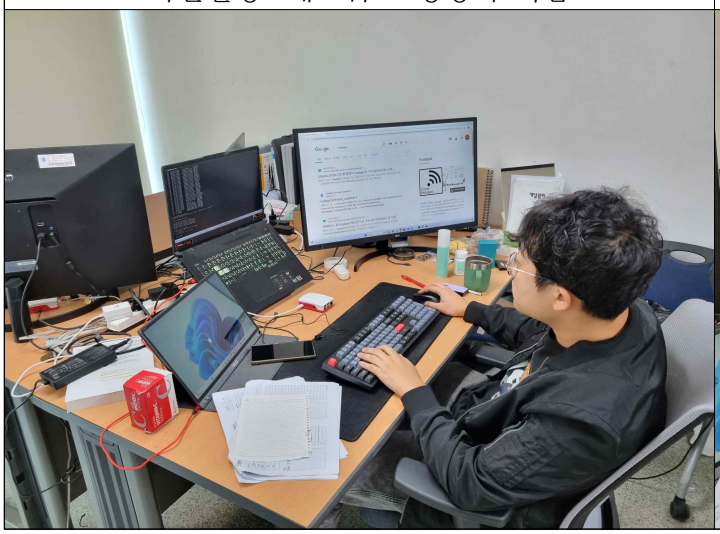
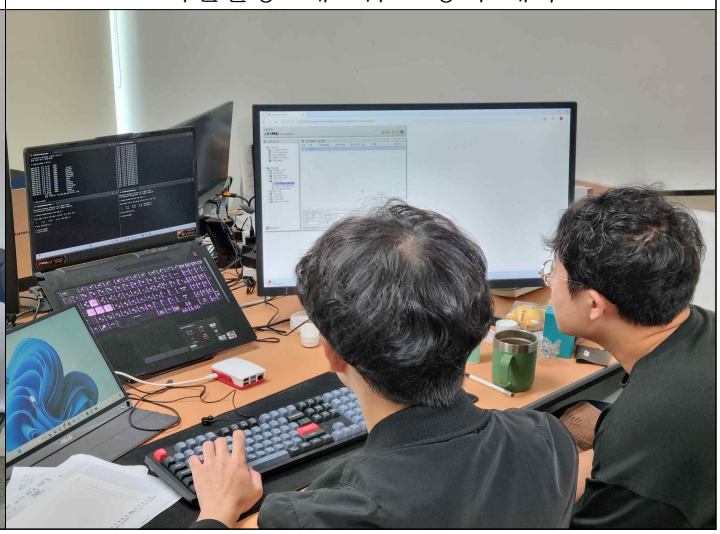
중간결과 및 소감

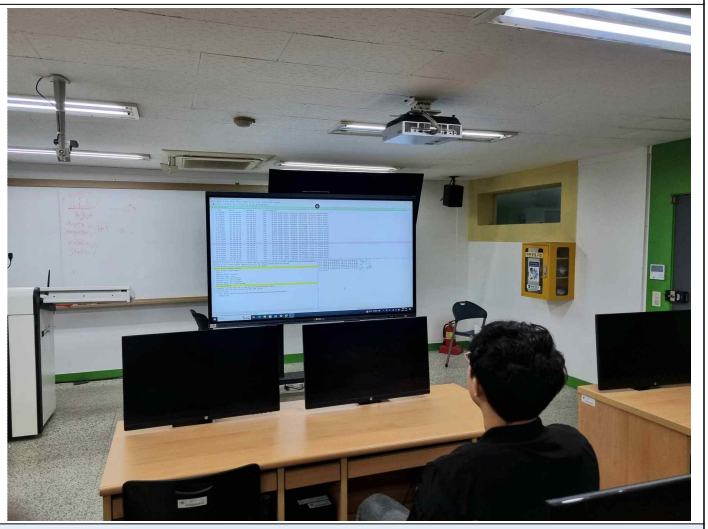
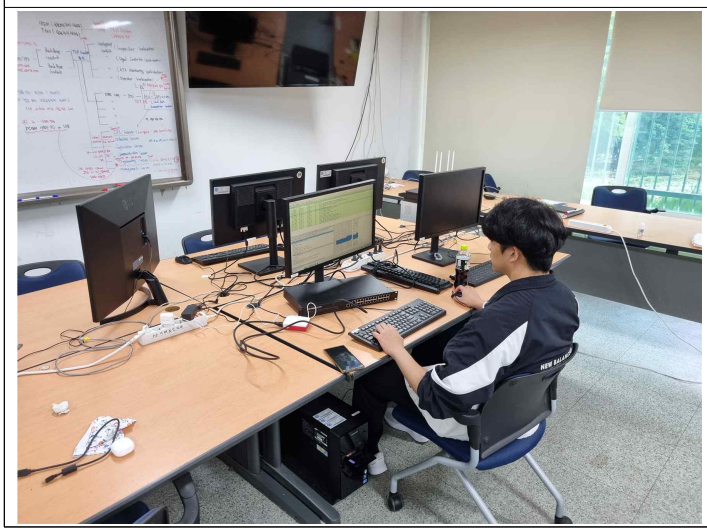
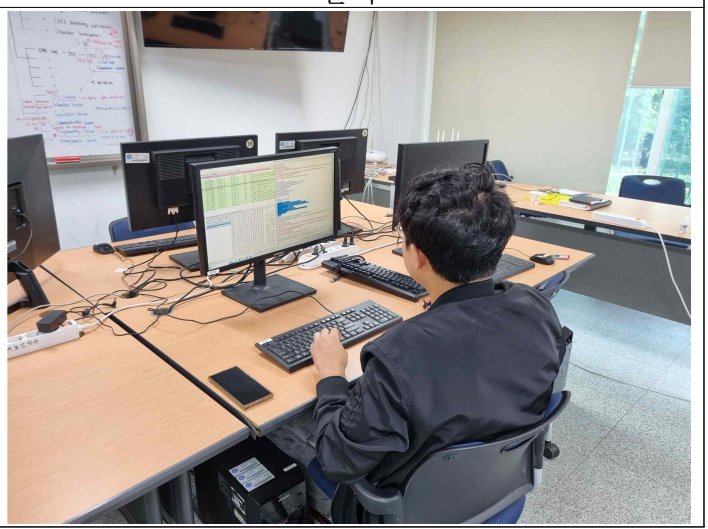
중간결과 : 네트워크 패킷 통신 과정 학습, 3 Way Handshake 및 4 Way Handshake 학습, 프로토콜 종류 및 학습, 라우터 기능 분석 및 보완(발전) 사항 파악, 네트워크 장비 제작, Wireshark 학습, 네트워크 위협 분석 체계(NTAS, Network Traffic Analysis System) 제작

소감: 이번 도전학기제를 통해 네트워크에 대한 기본 지식과 공유기의 세세한 기능을 연구하면서 많은 것을 알게 되어 뿌듯하게 생각한다. 평소에 웹 개발을 주로 해왔기 때문에 3-way handshake 과정을 깊게 살펴보지 않았고, 웹과 클라이언트 간의 통신이 잘 이루어지는지만 확인해왔다. 하지만 이번 기회에 Wireshark를 통해 그 과정들과 어떤 데이터들이 주고받는지를 알 수 있었다. 이를 통해 많은 것을 배웠고, 이 경험을 바탕으로 한층 더 성장한 나의 모습으로 도전학기제 과제를 마무리하고 네트워크 포트폴리오를 만들고 싶다.

	특히, 가장 흥미로웠던 점은 라우터의 기능이었다. 라우터는 네트워크 패킷의 경로를 지정해주는 역할을 하지만, 직접 구현할 일이 없어서 그 역할만 알고 나머지는 잘 몰랐다. 이번 프로젝트를 통해 실험하면서 라우터의 세세한 기능과 방화벽이 어떻게 작동하는지를 알게 되어 매우 좋았다. 마지막으로 남은 기간 동안 열심히 참여하여 좋은 성과를 이룰 수 있도록 노력하겠으며, 이 경험을 바탕으로 성장한 IT 전문가의 일원이 될 수 있도록 최선을 다하겠다.

[주차별 학습활동 사진]

<table>
<tr><th colspan="2">1주차</th></tr>
<tr><td>사진설명: TCP 통신 과정 학습</td><td>사진설명: 네트워크 패킷 구조 학습</td></tr>
<tr><td>  </td><td>  </td></tr>
<tr><th colspan="2">2주차</th></tr>
<tr><td>사진설명: 라우터 기능 분석</td><td>사진설명: 내 • 외부 네트워크 구축</td></tr>
<tr><td>  </td><td>  </td></tr>
<tr><th colspan="2">3주차</th></tr>
<tr><td>사진설명: 네트워크 명령어 학습</td><td>사진설명: 네트워크 장비 제작</td></tr>
<tr><td>  </td><td>  </td></tr>
</table>

<table>
<tr><th colspan="2">4주차</th></tr>
<tr><td>사진설명: 와이어샤크 학습</td><td>사진설명: 패킷 통신 과정 캡처 및 분석</td></tr>
<tr><td> 
</td><td>  </td></tr>
<tr><th colspan="2">5주차</th></tr>
<tr><td>사진설명: 스위치 패킷 캡처 및 분석</td><td>사진설명: 스위치 패킷을 캡처한 후 패킷 통신 과정 분석</td></tr>
<tr><td>  </td><td>  </td></tr>
</table>

[작성방법] ※ 제출 시 작성방법은 삭제 요망

1. 과제 수행 중간보고서는 과제 수행 계획서 및 주차별 활동보고서에 근거하여 상세히 작성
2. 프로젝트 수행사항
   가. 주요활동 내용은 최초 계획 대비 실제 활동내용 및 목표달성 여부를 상세히 기록
   나. **'개인'으로 참여할 경우 '개인목표 및 활동'으로 통합하여 작성**
   다. 투입시간: **3학점 기준 주당 9시간 이상** 활동
   라. 일시, 시간대별 활동 내용 등을 상세히 기록
3. 중간결과 및 소감
   가. 중간결과에 대해 책임교수의 지도를 받은 후 자유롭게 기술
   나. **10줄 이상 작성**, 활동사진, 도표 등 삽입 가능
4. 주차별 학습활동 사진(2장 이상) 첨부