**컴퓨터 네트워크 기말고사 KeyPoint**

**5장 네트워크 계층(Network Layer) 프로토콜**

네트워크계층 프로토콜은 경로 설정과 데이터그램의 순방향 전송 기능을 가진다.

IP 주소 공간의 소진 이유

1) 인터넷 상의 호스트 수의 증가

2) 클래스에 의한 주소 할당으로 주소공간의 낭비 초래

CIDR의 사용목적

1) 주소를 효율적으로 분배하기 위해

2) 라우터가 알아야 하는 라우팅 정보의 양을 최소화하려

IPv4 프로토콜의 특성으로 최선형 서비스, 비신뢰성, 비연결성, 비보안성을 들 수 있다.

IP 주소의 고갈 원인으로 인터넷 상의 호스트 수의 증가와 클래스에 의한 비효율적인 주소 할당을 들 수 있다.

IP 데이터그램은 헤더와 페이로드 부분으로 이루어지며, 헤더는 라우팅 정보를 가지며, 페이로드에는 일반적으로 세그먼트가 캡슐화된다. 헤더의 크기는 20 ~ 60 바이트로 가변이다.

단편화를 지원하기 위하여 IP헤더에는 식별자 필드와 플래그 그리고 단편화 옵션 필드가 사용되어진다.

단편화된 데이터그램에 대한 재조립은 최종 목적지 호스트에서만 이루어진다. ( ~ 중간고사 범위)

링크 상태 알고리즘: 노드들은 인접한 링크들에 대한 상태 정보를 포함하는 링크 상태 패킷을 교환함으로 네트워크 토폴로지와 노든 링크 비용을 가지고 라우팅을 시행한다.

Dijkstra’s 알고리즘: 링크 상태 라우팅 알고리즘에 사용되며, 중앙 집중 라우팅 으로서 최소비용 경로를 찾는다

RIP: RIP는 주어진 네트워크에서 가장 적은 수의 라우터를 가지는 경로를 찾는 인트라-AS 라우팅 프로토콜이다.

OSPF: 링크 상태 알고리즘을 사용하는 인트라-AS 라우팅 프로토콜이다.

BGP: 특정 정책(policy)를 고려하여 라우팅 기능을 수행하는 인터-AS 라우팅 프로토콜로서 경로 벡터 알고리즘을 사용한다.

IP 프로토콜은 오류 메시지를 보낼 때 ICMP를 사용하고, ICMP는 메시지를 전달하기 위해 IP를 사용한다.

**6장 전송계층(Transport Layer) 프로토콜**

전송계층에서는 서로 다른 두 호스트에서 동작하고 있는 프로세스와 프로세스간의 전송을 책임진다.

트랜스포트 계층에서 프로세스-프로세스 전송을 위해 <IP 주소, 포트 번호> 쌍이 필요하다.

수신 호스트의 전송계층 프로토콜이 네트워크 계층으로부터 수신한 세그먼트를 해당 응용 프로세스들에게 분배하여 보내는 일을 역다중화(demultiplexing)라고 한다.

소스 호스트에서 여러 응용 프로세스로부터 데이터를 모아서 헤더를 첨가하여 세그먼트로 만들어서 네트워크 계층으로 세그먼트를 보내는 일을 다중화(multiplexing)라 부른다.

전송계층에서는 프로세스-프로세스 전송과 자름 재조립 기능(Segment & Reassembly: SAR), 그리고 다중화와 역다중화 지원한다.

포트 번호는 하나의 호스트 상에 동작하는 여러 프로세스들을 구분하는데 사용되어진다.

TCP 프로토콜은 신뢰성 있는 전송을 지원하는데, 이를 위하여 Ack, 타이머, 재전송, 순서번호 등의 메커니즘을 가진다.

TCP 프로토콜의 PDU를 세그먼트(Segment)라 부르는데, 20~60 바이트 크기의 헤더와 페이로드(데이터)로 이루어진다.

TCP 헤더의 수신 윈도우(rcvwind) 필드는 흐름제어에 사용된다.

TCP 연결은 3-way 핸드셰이크 과정을 통하여 설정된다.

TCP 연결 설정과정에서 SYN, ACK 비트가 사용된다.

TCP 연결 해지 과정에서 FIN 비트가 사용되어진다.

TCP의 타임아웃은 RTT에 기반으로 설정된다.

재전송 타이머가 타임아웃인 경우에 소스 호스트는 전송한 패킷이 손실되었다고 간주하고 재전송한다.

흐름제어는 수신 호스트의 여유 버퍼 크기에 의존한다.

슬라이딩 윈도우 흐름제어에서 송신 호스트의 윈도우의 크기는 Ack를 수신한 경우에는 윈도우의 좌측이 줄어들고, 새로운 세그먼트를 전송하면 윈도우의 우측이 늘어난다.

패킷교환 네트워크에서 여러 사용자가 네트워크 자원을 공유하기 때문에, 네트워크내에서 혼잡이 발생할 수 있다. 이러한 혼잡이 발생하게 되면, 데이터 전송에서의 지연(delay)이 발생하게 되고, 심하면 라우터에서의 버퍼 오버플로우로 인한 패킷의 손실(loss)이 발생할 수 있다.

TCP의 혼잡제어는 네트워크 내의 라우터의 도움 없이 소스 호스트와 목적지 호스트 사이에 Ack를 주고받음으로써 종단간 호스트 사이에서 이루어진다.

슬로우 스타트 단계에서는 처음에 1개의 패킷을 전송하고, 임계 값에 도달할 때까지 매 전송에서 전송되는 패킷의 수를 2배로하여 전송한다.

혼잡회피 과정에서 왕복시간 마다 윈도우의 크기를 1씩 증가시킨다. 패킷의 손실이 발생하면 다음의 전송 윈도우의 크기는 1로 하고 임계 값은 손실이 발생하였을 때의 윈도우 크기의 반으로 만든다.

**빠른 재전송(fast retransmission):** 소스 호스트는 전송된 패킷에 대하여 연속된 3개의 중복(duplicated) 패킷을 수신하는 경우에, 그 패킷이 네트워크 혼잡에 의해 손실된 것으로 간주하고 빠르게 재전송한다. 이렇게 중복된 Ack를 통하여 빠르게 재전송함으로써 전송 효율을 높일 수 있다.

**빠른 회복(fast recovery):** 전송된 패킷의 손실이 발생할 경우, 혼잡 윈도우의 크기를 손실 발생 당시의 윈도우 크기의 반으로 줄인 후에 혼잡회피 과정을 통하여 패킷을 전송함으로, 보다 빠른 혼잡윈도우의 크기를 크게 한다. 이러한 과정을 통하여 보다 나은 전송 효율을 높일 수 있다.

TCP 소스의 실재 윈도우의 크기는 수신자로부터 받은 흐름제어 윈도우와 혼잡제어 윈도우 중에서 작은 값으로 삼는다.

UDP 프로토콜은 전송계층 프로토콜이 가지는 가장 기본적인 기능만 가지기 때문에 헤더가 간단하다.

UDP는 오버헤드가 적고 매우 빠른 전달 서비스를 제공하는 신뢰성 없고 비연결성인 프로토콜로서, 적은 데이터를 주고받는 DNS, SNMP 등과 같은 요청-응답 형의 서비스 응용과 어느 정도의 손실을 허용하는 멀티미디어 응용에 적합한 전송 계층 프로토콜이다.

**7장 응용계층 프로토콜**

소켓은 호스트 내의 응용계층과 전송계층 사이의 인터페이스이다.

호스트 상에서 동작하는 프로세스를 인식하기 위하여서 호스트의 이름 혹은 인터넷 주소와 프로세스를 나타내는 포트 번호가 필요하다.

네트워크 응용 개발자들은 자신의 응용이 요구하는 서비스에 맞는 전송계층 프로토콜인 TCP와 UDP 중에 하나를 선택해야 한다.

웹 응용을 위한 응용계층 프로토콜 HTTP는 전송계층의 프로토콜로 TCP를 사용한다.

비지속(non-persistent) 연결전송 방식에서는 하나의 파일 전송을 위하여 하나의 TCP 연결이 필요하고, 파일 전송 후에 TCP 연결이 해지된다. 그러나 지속 연결 전송 방식에서는 처음에 설정된 TCP 연결을 통하여 모든 객체들이 전송된다.

인증 메커니즘은 서버에서 클라이언트에게 사용자 이름(ID)과 비밀번호(password)를 요구하는 것을 말한다.

쿠키는 사이트가 사용자를 추적하는 또 다른 방법으로써 포탈, 전자상거래, 광고 사이트들에서 주로 사용되어진다.

SMTP는 메일을 송신자의 메일 서버로부터 수신자의 메일 서버에게 전송하는데 전송계층 프로토콜로 TCP를 사용한다.

SMTP 클라이언트로부터 전송되어진 메시지는 절대로 중간의 메일서버에 저장되지 않는다.

MIME은 SMTP를 통하여 ASCII 가 아닌 멀티미디어 데이터를 전송할 수 있도록 하는 SMTP의 확장판이다.

FTP는 클라이언트 호스트와 서버 호스트 사이에 파일 전송을 위해 두 개의 TCP 연결을 설정하는데, 하나는 데이터 전송을 위하여 사용되고, 나머지 하나는 명령과 응답 등의 제어 정보를 위하여 사용되어진다.

호스트 이름과 IP 주소와의 변환을 대신해 주는 서비스가 바로 인터넷 DNS(Domain Name System)의 주 임무이다.