



# BoostCamp iOS

MOGAY

# 순서

---

- Bit 자리수 문제
- IPv6?
- IPv6에 대한 애플 정책

# Bit 자리수 문제

---

- 소수점 표현 문제
  - 고정 소수점 -> 부동 소수점
- 32bit 자릿수 overflow
  - 32bit로 데이터의 표현
  - psy 뮤직비디오 오버플로우
  - 2038년 1월 19일 문제
- 해결?
  - 32bit -> 64bit

# IPv6 ??

---

현재 IPv4의 문제점인 주소고갈, 보안성, 이동성 지원 등을 해결하기 위해 개발된, 128 bit 주소체계를 갖는 차세대 인터넷의 핵심 프로토콜

0032:0000:0000:0752:0DB8:5123:2068:3320

$2^{32}$   
(약 40억)



$2^{128}$   
( $3.4 \times 10^{38}$ )

# IPv6 특성

---

IP 주소 확장

➡ 약 42억 → 약 31조

호스트 주소 자동 설정

➡ 네트워크 관리자로부터 IP 주소를 부여 받아 수동으로 설정 → 자동으로 주소 부여

패킷 크기 확장

➡ 64킬로바이트로 제한 → (점보그램 옵션) 특정 호스트 사이에 교환 패킷 크기 제한 없음

효율적인 라우팅

➡ 신속한 IP 패킷 처리와 네트워크 기능 확장을 위해 고정크기 헤더와 확장 헤더 동시 사용

# IPv6 특징

---

플로 레이블링 (Flow Labeling)

➡ 특정 트래픽은 별도의 특별한 처리를 통해 높은 품질의 서비스 제공

인증 및 보안 기능 (IPSec)

➡ 패킷 출처 인증과 데이터 무결성 및 비밀 보장 기능 추가  
(IPv6 확장헤더)

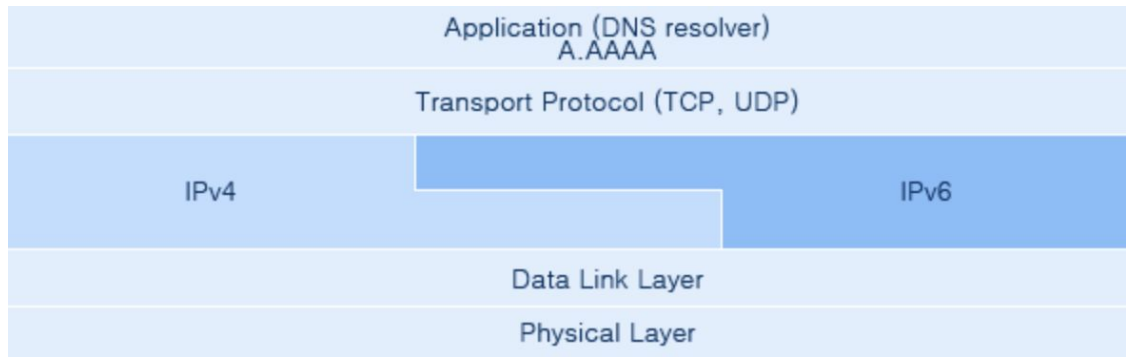
이동성 (모바일)

➡ 네트워크의 물리적 위치에 제한 받지 않고 주소 유지 및 이동 가능

# IPv6 - Dual Stack

---

하나의 시스템에서 IPv4와 IPv6 프로토콜 모두 처리



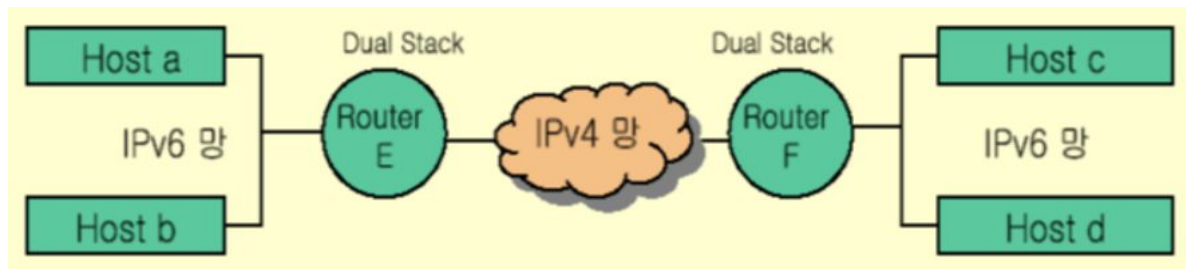
장점 : DNS 주소 해석 라이브러리가 두 유형의 IP주소 유형 모두 지원

단점 : 프로토콜 스택 수정으로 과도한 비용 발생

# IPv6 - Tunneling

---

서로 다른 프로토콜 사이의 중간 프로토콜에서 사용하는 프로토콜로 캡슐화



종류 : 설정 터널링, 자동 터널링

장점 : 표준화 활동이 가장 활발, 다양한 터널링 기술이 표준으로 제안됨

단점 : 구현이 어렵고 복잡한 동작과정



# IPv6 - Address Management

---

IPv4, IPv6 사이에 주소 변환기를 사용하여 상호 연동

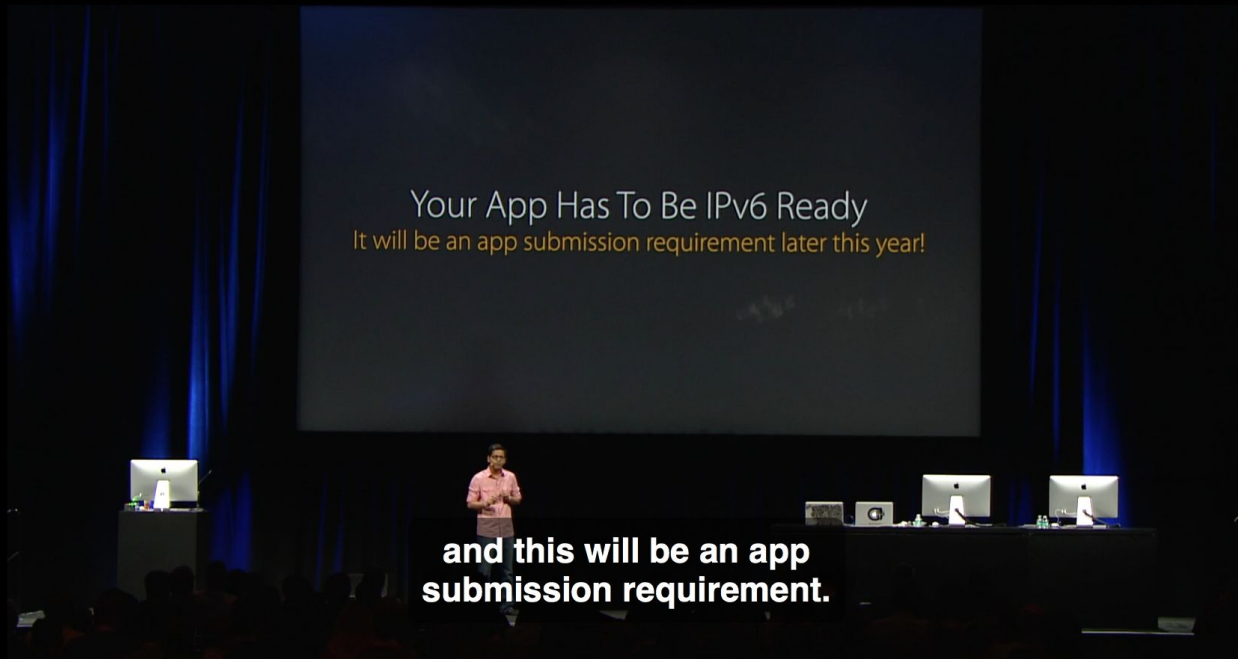


종류 : 헤더 변환 방식, 전송 계층 릴레이 방식, 응용 계층 게이트웨이 방식

장점 : DNS 주소 해석 라이브러리가 두 유형의 IP주소 유형 모두 지원

단점 : 포트콘 스테에 대해 수정이 필요 있으며 변환방식이 특명 구현이

2015.06



**and this will be an app  
submission requirement.**

**WWDC 2015** Your App and Next Generation Networks

# Supporting IPv6-only Networks

— — —

## Supporting IPv6-only Networks

May 4, 2016

At WWDC 2015 we announced the transition to IPv6-only network services in iOS 9. Starting June 1, 2016 all apps submitted to the App Store must support IPv6-only networking. Most apps will not require any changes because IPv6 is already supported by NSURLSession and CFNetwork APIs.

If your app uses IPv4-specific APIs or hard-coded IP addresses, you will need to make some changes. Learn how to ensure compatibility by reading [Supporting IPv6 DNS64/NAT64 Networks](#) and watch [Your App and Next Generation Networks](#).



<https://developer.apple.com/support/ipv6/>

**Starting June 1, 2016, all apps submitted to the App Store must support IPv6-only networking.**

2016.06.01

# Apple의 IPv6에 대한 정책

---



**Apple  
Rejected  
My App!**

# We Should

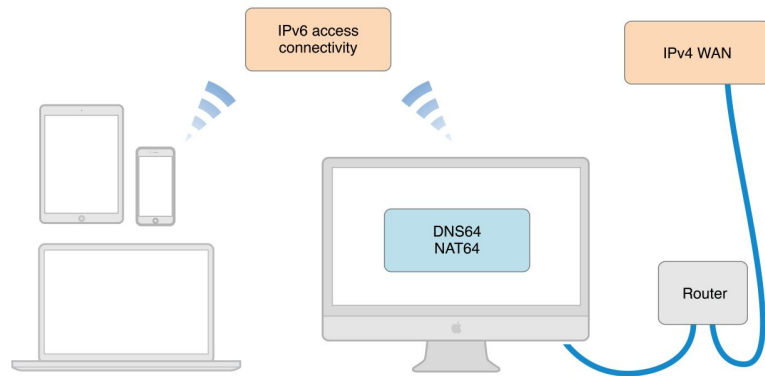
---

- IP Literal (hard-coded IP address)
  - “11.22.33.44”, “0.0.0.0” (Reachability Test:Network Preflight)
  - cf) host name (FQDN:Fully Qualified Domain Name)
- IP Address를 설정파일이나 서버로부터 획득하여 Socket연결하는지
- IP Address를 저장하기 위하여 32bit이하 크기의 변수타입
  - uint32\_t, in\_addr, sockaddr\_in
- Low-Level Networking API 사용
  - inet\_addr(), inet\_aton(), inet\_lnaof(), inet\_makeaddr() ...
- 이러한 것들을 사용하는 라이브러리를 사용x
  - 일반적으로 네트워크 통신 시, NSURLSession과 CFNetwork 같은 High Level API의 경우는 이미 IPv6에 대한 지원에 문제가 없으나, Low-Level socket APIs를 직접 사용할 경우에는 구조체와 함수 등 IPv4 전용 함수를 사용하고 있는지에 대해서 확인이 필요합니다.

# Testing your app in an IPv6-only environment

---

You should test your app on an IPv6-only network. If you don't have one, you can set up a test network by following the instructions in [Test for IPv6 DNS64/NAT64 Compatibility Regularly](#)



For technical issues encountered while transitioning to IPv6, see:

- [Apple Developer Forums](#)
- [Developer Technical Support](#)

`https://developer.apple.com/support/ipv6/`