**Dynamic Host Configuration Protocol**

March 1997

# 목차

[목차 2](#_Toc190183965)

[1. Introduction 2](#_Toc190183966)

[1.1 Motivation 3](#_Toc190183967)

[1.2 Requirements Terminology 4](#_Toc190183968)

[1.3 Applicability Statement 4](#_Toc190183969)

[1.4 Relation to RFC 4253 5](#_Toc190183970)

[1.5 The NETCONF/RESTCONF Convention 5](#_Toc190183971)

[2. Solution Overview 6](#_Toc190183972)

[3. The NETCONF or RESTCONF Client 7](#_Toc190183973)

[3.1 Client Protocol Operation 8](#_Toc190183974)

[3.2 Client Configuration Data Model 9](#_Toc190183975)

[4. The NETCONF or RESTCONF Server 9](#_Toc190183976)

[4.1 Server Protocol Operation 10](#_Toc190183977)

[4.2 Server Configuration Data Model 11](#_Toc190183978)

[5. Security Considerations 11](#_Toc190183979)

[6. IANA Considerations 14](#_Toc190183980)

[7. References 16](#_Toc190183981)

[9.1 Normative References 16](#_Toc190183982)

[9.2 Informative References 17](#_Toc190183983)

[Appendix A. Changes from RFC 4742 17](#_Toc190183984)

# 1. Introduction

**✅ DHCP란?**

DHCP(Dynamic Host Configuration Protocol)는 **인터넷에 연결되는 기기(호스트)들이 자동으로 네트워크 설정을 받을 수 있도록 도와주는 프로토콜**이야.  
쉽게 말해, **IP 주소를 자동으로 할당하고 필요한 설정 정보를 제공하는 시스템**이라고 보면 돼.

**✅ DHCP는 어떻게 동작할까?**

DHCP는 **클라이언트-서버 모델**을 기반으로 동작해.

* **서버(DHCP Server)**: 네트워크 설정 정보를 제공하는 역할.
* **클라이언트(Client)**: 네트워크 설정을 요청하는 장치(PC, 스마트폰, 프린터 등).

예를 들어, 네트워크에 새로 연결된 컴퓨터가 있다면, DHCP 서버에 "IP 주소 주세요!"라고 요청하고, DHCP 서버가 적절한 IP 주소를 할당해주는 방식이야.

👉 **중요한 점**: DHCP 서버 역할은 네트워크 관리자가 지정한 특정 장치에서만 수행해야 해.  
아무 장치나 DHCP 서버처럼 동작하면 네트워크 충돌이 발생할 수 있기 때문이야.

**✅ DHCP의 IP 주소 할당 방식**

DHCP는 세 가지 방법으로 IP 주소를 할당할 수 있어.

1. **자동 할당 (Automatic Allocation)**
   * 한 번 할당한 IP 주소를 영구적으로 유지하는 방식.
   * 고정된 장치(예: 서버, 네트워크 프린터)에 적합.
2. **동적 할당 (Dynamic Allocation)**
   * 일정 시간 동안만 IP 주소를 할당한 후, 필요 없으면 회수하는 방식.
   * 많은 기기가 연결되고 끊어지는 환경(예: Wi-Fi 네트워크)에 적합.
   * IP 주소가 부족한 경우, 사용하지 않는 주소를 다시 활용 가능.
3. **수동 할당 (Manual Allocation)**
   * 네트워크 관리자가 미리 지정한 IP 주소를 특정 장치에 할당.
   * DHCP는 단순히 해당 주소를 전달하는 역할만 수행.
   * 중요한 네트워크 장비(예: 서버, 라우터)에 사용될 수 있음.

👉 **"동적 할당" 방식이 가장 효율적!**  
특히 IP 주소가 한정된 네트워크에서는 **더 이상 사용하지 않는 IP를 회수해서 다른 장치에 재사용할 수 있기 때문에** 유용해.

**✅ DHCP는 왜 BOOTP를 기반으로 만들어졌을까?**

DHCP는 기존에 있던 **BOOTP (Bootstrap Protocol)**를 확장해서 개발되었어. BOOTP와 호환되도록 설계되어 있어서, 기존의 BOOTP 장치도 DHCP 서버와 함께 사용할 수 있어.

또한, **BOOTP 릴레이 에이전트**를 활용하면, 모든 네트워크에 DHCP 서버를 둘 필요 없이, 다른 네트워크에 있는 DHCP 서버에서 IP를 받을 수도 있어.

**✅ 정리하면?**

✔ DHCP는 **네트워크 장치들이 자동으로 IP 주소를 받을 수 있도록 도와주는 프로토콜**이야.  
✔ DHCP 서버가 클라이언트에게 **IP 주소와 기타 네트워크 설정 정보를 제공**해.  
✔ \*\*세 가지 IP 할당 방식(자동, 동적, 수동)\*\*이 있으며, **동적 할당이 가장 효율적**.  
✔ **BOOTP 기반으로 만들어졌으며**, 기존 BOOTP 클라이언트와도 호환 가능.  
✔ DHCP 릴레이 기능을 활용하면 **모든 네트워크에 DHCP 서버를 따로 둘 필요 없이 중앙에서 관리 가능**.

👉 **결론: DHCP 덕분에 우리가 복잡한 네트워크 설정 없이 인터넷을 쉽게 사용할 수 있는 거야!**

## 1.1 Changes to RFC 1541

**RFC 1541에서 변경된 사항**

RFC 2131은 이전 DHCP 표준인 **RFC 1541**을 개선한 버전.

**✅ 1. 새로운 DHCP 메시지 타입 추가 (DHCPINFORM)**

* **DHCPINFORM 메시지**가 새롭게 추가되었어.
* 이 메시지는 클라이언트가 **IP 주소 할당 없이 추가적인 설정 정보만 요청할 때** 사용돼.
  + 예를 들어, **이미 IP 주소를 갖고 있는 장치가 DHCP 서버에서 추가적인 네트워크 정보를 받고 싶을 때** 활용할 수 있어.
* 자세한 내용은 **섹션 3.4, 4.3, 4.4**에서 설명됨.

**✅ 2. DHCP 클라이언트 식별 방식 확장 ("벤더 클래스" 추가)**

* 기존에는 DHCP 서버가 **클라이언트를 구분하는 방식이 제한적**이었어.
* 이제 **벤더 클래스(Vendor Class)** 개념이 추가되어,
  + 특정 제조사(예: Cisco, HP)의 장비인지
  + 어떤 운영 체제(예: Windows, Linux)인지
  + 특정한 기능을 가진 장비인지  
    **더 정교하게 식별 가능해.**
* 관련 내용은 **섹션 4.2, 4.3**에서 설명돼.

**✅ 3. 최소 임대 기간(Minimum Lease Time) 제한 제거**

* 이전에는 **DHCP 서버가 IP를 할당할 때 최소한의 임대 기간이 정해져 있었어.**
* 하지만 **이 제한이 없어지면서 더 유연한 IP 주소 할당이 가능**해졌어.
  + 예를 들어, 어떤 네트워크에서는 **아주 짧은 시간 동안만 IP 주소를 할당**할 수도 있음.
  + 반대로, 특정 장비에 **오랫동안 같은 IP 주소를 유지**할 수도 있음.

**✅ 4. 문서 내용 개선 (더 명확하게 정리됨)**

* DHCP의 **상호 운용성 테스트**(DHCP 장비 간의 호환성을 시험하는 과정) 중 발견된 문제점을 반영하여,
* **설명을 더 명확하게 정리하고 문장을 다듬었어.**

✅ 한눈에 보는 RFC 1541 → RFC 2131 변경 사항 정리

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

👉 **결론**: RFC 2131은 DHCP를 더 유연하고 실용적으로 개선한 버전이야!

## 1.2 Related Work

**관련 기술 및 연구**

DHCP와 비슷하거나 관련된 기술들이 이미 여러 개 존재해. 이 섹션에서는 **DHCP와 관련된 기존 기술**들을 소개하고 있어.

**✅ 1. 기존의 네트워크 주소 할당 기술**

과거에는 네트워크에서 장치가 **IP 주소를 자동으로 받는 방식**이 다양했어.

1️ **RARP (Reverse Address Resolution Protocol)**

* 네트워크 카드(MAC 주소)를 가지고 **해당 장치의 IP 주소를 찾는 방법**이야.
* **단점:** IP 주소만 찾을 뿐, 추가적인 설정 정보(서브넷 마스크, 게이트웨이 등)는 제공하지 못해.
* 이후 확장된 버전인 \*\*DRARP (Dynamic RARP)\*\*에서 자동 IP 할당 기능이 추가되었어.

2️ **BOOTP (Bootstrap Protocol)**

* DHCP의 **이전 버전**이라고 보면 돼.
* 클라이언트가 부팅할 때 서버로부터 **IP 주소와 추가 설정 정보(게이트웨이, DNS 등)를 받을 수 있어.**
* **단점:** IP 주소를 동적으로 회수(reuse)하는 기능이 없어서 관리가 어려웠어.
* BOOTP는 확장이 가능해서, 나중에 여러 기능이 추가되었고 DHCP의 기초가 되었어.

3️ **NIP (Network Information Protocol)**

* MIT 아테나 프로젝트에서 사용되던 방식이야.
* **분산 네트워크 환경에서 동적으로 IP 주소를 할당하는 방법**을 연구했어.
* DHCP의 아이디어와 비슷하지만, 널리 사용되지는 않았어.

**✅ 2. 장치의 부팅과 네트워크 설정 관련 기술**

IP 주소만 할당하는 것이 아니라, 네트워크 장치가 제대로 동작하려면 **운영체제, 설정 파일, 라우터 정보** 등도 필요해. 이를 돕는 여러 기술이 있었어.

1️ **TFTP (Trivial File Transfer Protocol)**

* 클라이언트가 **운영체제(OS) 파일을 서버에서 다운로드**할 수 있도록 도와주는 프로토콜이야.
* 주로 \*\*디스크 없는 장치(diskless workstation)\*\*에서 사용돼.
* 예: 네트워크 장비(스위치, 공유기 등)가 부팅할 때 펌웨어를 서버에서 가져오는 경우.

2️ **ICMP (Internet Control Message Protocol)**

* 네트워크에서 **라우터를 찾거나 추가적인 정보를 얻을 때 사용**되는 프로토콜이야.
* 예를 들어:
  + **ICMP Redirect 메시지**: 더 나은 경로가 있을 때 알려줌.
  + **ICMP Mask Request 메시지**: 서브넷 마스크 정보를 요청할 때 사용.
  + **ICMP Router Discovery**: 네트워크에서 라우터를 자동으로 찾을 수 있도록 도와줌.

3️ **Sun Microsystems의 "bootparams"**

* Sun Microsystems에서 개발한 **디스크 없는 워크스테이션**을 위한 부팅 기술이야.
* RARP + TFTP + bootparams(설정 파일 전달)를 조합하여, OS를 다운로드하고 설정하는 방식이었어.
* 일부 Sun 네트워크에서는 DRARP와 자동 설치(auto-installation) 기능을 이용해서 새로운 장치를 자동 설정하기도 했어.

**✅ 3. 기타 관련 기술**

1️ **MTU (Maximum Transmission Unit) 경로 탐색 알고리즘**

* 네트워크에서 **최대 패킷 크기(MTU)를 자동으로 탐색**하는 기술이야.
* 너무 큰 패킷을 보내면 잘리지 않고 효율적으로 전송될 수 있도록 도와줘.

2️ **ARP (Address Resolution Protocol)의 확장 가능성**

* ARP는 보통 **IP ↔ MAC 주소 변환**에 사용되지만,
* **네트워크 리소스를 찾는 용도로 확장할 수도 있다**는 연구가 있었어.

3️ **Host Requirements RFCs (RFC 3, RFC 4)**

* 네트워크 장치(호스트)가 초기 설정을 할 때 필요한 요구 사항을 정리한 문서야.
* 특히 **디스크 없는 장치의 네트워크 설정 방법**을 제안하고 있어.

✅ 정리: DHCP와 관련된 기존 기술들

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

👉 **결론:** DHCP는 기존의 여러 기술들(BOOTP, RARP, ICMP 등)의 단점을 개선하고, **동적 IP 할당 및 네트워크 설정을 더 쉽게 만들기 위해 탄생**한 프로토콜이야!

## 1.3 Problem definition and issues

DHCP는 네트워크에 연결되는 장치(클라이언트)들에게 **필요한 설정 정보를 자동으로 제공하는 역할**을 해. 이 섹션에서는 **DHCP가 해결하는 문제와 한계점**을 설명하고 있어.

**✅ 1. DHCP의 목표**

DHCP는 **Host Requirements RFCs**(네트워크 장치가 인터넷에서 정상적으로 동작하기 위해 필요한 설정 목록)에 따라 클라이언트가 필요한 네트워크 설정을 자동으로 받도록 설계되었어.

즉, **DHCP를 통해 설정 정보를 받은 클라이언트는 인터넷의 다른 모든 장치와 정상적으로 통신할 수 있어야 해.**

👉 **DHCP가 제공하는 주요 네트워크 설정 정보** (Appendix A에 자세히 나옴)

* IP 주소
* 서브넷 마스크
* 기본 게이트웨이
* DNS 서버 주소
* 기타 TCP/IP 관련 설정

**✅ 2. 클라이언트가 필요한 설정만 받을 수 있음**

* 모든 설정 정보가 **필수적인 것은 아님.**
* 클라이언트가 필요한 설정만 선택적으로 받을 수 있음.
* DHCP 서버와 클라이언트가 협상해서 **특정 서브넷(네트워크 구역)에 필요한 정보만 전송 가능.**

👉 예를 들어:

* 어떤 클라이언트는 **DNS 서버 정보**만 필요할 수도 있음.
* 어떤 서브넷에서는 **특정 게이트웨이 정보**만 필요할 수도 있음.

**✅ 3. DHCP가 하지 않는 것 (한계점)**

1️ **IP와 직접 관련 없는 설정은 지원하지 않을 수도 있음**

* DHCP는 기본적으로 **TCP/IP 관련 설정 정보만 제공**해.
* 하지만 네트워크 관리자가 원하면 다른 설정 정보를 추가할 수도 있음.

2️ **DNS(도메인 네임 시스템)와의 연동이 자동으로 되지는 않음**

* 클라이언트가 DHCP를 통해 **IP 주소를 할당 받더라도**,
* 이 IP 주소를 **DNS에 자동 등록하는 기능은 DHCP가 직접 처리하지 않음.**
* (일부 네트워크에서는 추가적인 설정으로 DNS 등록을 자동화할 수도 있음.)

3️ **라우터 설정은 DHCP가 담당하지 않음**

* **DHCP는 클라이언트 장치(PC, 스마트폰, IoT 기기 등)만 설정**할 수 있어.
* **라우터 자체의 설정을 바꾸는 용도로 설계되지 않았음.**

👉 **예를 들어:**

* PC, 스마트폰, 태블릿: DHCP를 통해 자동으로 네트워크 설정 받음 ✅
* 네트워크 라우터, 스위치: 직접 설정해야 함 ❌

✅ 결론: DHCP의 역할과 한계

텍스트, 폰트, 라인, 번호이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

💡 **정리하자면:**

* DHCP는 **클라이언트 장치가 네트워크를 쉽게 사용할 수 있도록 IP 주소 및 설정을 자동 제공**하는 역할을 해.
* 하지만 **DNS 자동 등록이나 라우터 설정은 DHCP의 역할이 아니며, 추가적인 설정이 필요할 수도 있음!**

## 1.4 Requirements

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

## 1.5 Terminology

**✅ 1. "DHCP 클라이언트 (DHCP client)"**

📌 **뜻:**

* **네트워크에 연결될 때 DHCP 서버에서 설정 정보를 받는 장치**
* **IP 주소, 서브넷 마스크, 게이트웨이 주소** 등을 DHCP 서버에서 자동으로 할당 받음.

📌 **예제:**

* **노트북, 스마트폰, PC** 등 네트워크에 접속하는 모든 장치가 DHCP 클라이언트가 될 수 있어.
* 집에서 와이파이에 연결할 때, IP 주소를 자동으로 받는 장치들은 DHCP 클라이언트야.

💡 **쉽게 말하면:**  
👉 "**네트워크 설정 정보를 받는 기기 (예: 노트북, 스마트폰 등)**"

**✅ 2. "DHCP 서버 (DHCP server)"**

📌 **뜻:**

* **DHCP 클라이언트에게 네트워크 설정 정보를 제공하는 장치**
* 클라이언트가 요청하면 **IP 주소, 서브넷 마스크, DNS 서버 주소** 등을 자동으로 할당해 줌.

📌 **예제:**

* **집에서 사용하는 인터넷 공유기 (Wi-Fi 라우터)** → 보통 DHCP 서버 역할을 함.
* **회사 네트워크 서버** → 여러 클라이언트들에게 IP 주소를 자동 할당.

💡 **쉽게 말하면:**  
👉 "**네트워크 설정 정보를 제공하는 서버 (예: 공유기, 네트워크 서버 등)**"

**✅ 3. "BOOTP 릴레이 에이전트 (BOOTP relay agent)"**

📌 **뜻:**

* **DHCP 클라이언트와 DHCP 서버 사이에서 메시지를 전달하는 중계 장치**
* 클라이언트와 서버가 **같은 네트워크에 없을 때**, 릴레이 에이전트가 DHCP 요청을 서버로 전달해 줌.

📌 **예제:**

* 큰 회사에서 **여러 개의 네트워크가 있을 때**, DHCP 서버가 모든 네트워크에 직접 연결되어 있지 않으면 **릴레이 에이전트가 중간에서 연결을 도와줌**.
* 공유기가 DHCP 서버 역할을 하지 않고, **인터넷 제공업체(ISP) 서버에서 IP 주소를 받아오는 경우** 릴레이 에이전트가 필요할 수도 있음.

💡 **쉽게 말하면:**  
👉 "**DHCP 클라이언트와 서버 사이에서 메시지를 전달하는 중계 장치**"

**✅ 4. "바인딩 (Binding)"**

📌 **뜻:**

* **DHCP 서버가 특정 클라이언트에게 할당한 네트워크 설정 정보 모음**
* 최소한 **IP 주소는 반드시 포함**되며, 다른 설정 정보(서브넷 마스크, DNS 주소 등)도 함께 저장될 수 있음.
* 바인딩 정보는 DHCP 서버가 관리하며, 특정 시간이 지나면 갱신되거나 해제될 수 있음.

📌 **예제:**

* DHCP 서버가 클라이언트에게 **192.168.1.100**이라는 IP 주소를 할당하면, 이 정보는 바인딩 데이터에 저장됨.
* 만약 클라이언트가 일정 시간 동안 네트워크를 사용하지 않으면, DHCP 서버는 이 IP 주소를 다른 클라이언트에게 할당할 수도 있음.

💡 **쉽게 말하면:**  
👉 "**DHCP 서버가 특정 클라이언트에게 할당한 네트워크 설정 정보 모음 (IP 주소 포함)**"

## 1.6 Design goals

**설계 목표**

**✅ 1. DHCP는 "정책"이 아니라 "기능"이어야 한다**

* DHCP는 단순히 **네트워크 설정 정보를 제공하는 도구(기능)** 역할만 해야 함.
* 특정 정책(예: 누가 IP를 받을 수 있는지, 얼마나 오래 유지되는지 등)은 **시스템 관리자**가 결정해야 함.

**✅ 2. 클라이언트는 수동 설정이 필요 없어야 한다**

* DHCP를 사용하는 **모든 장치(클라이언트)는 사람이 직접 설정할 필요 없이 자동으로 네트워크 정보를 받아야 함.**
* 사용자 개입 없이 IP 주소, 서브넷 마스크, 게이트웨이 등을 자동 설정해야 함.

**✅ 3. 네트워크 관리자가 개별 클라이언트를 일일이 설정하지 않아도 돼야 한다**

* DHCP를 사용하면 네트워크 관리자가 **각 클라이언트(PC, 스마트폰 등)의 IP 설정을 수동으로 입력할 필요 없음**.
* 네트워크에 새로운 장치가 연결되면, **DHCP가 자동으로 설정 정보를 제공해야 함**.

**✅ 4. 모든 서브넷에 DHCP 서버가 있을 필요는 없어야 한다**

* **서브넷(Subnet)**: 네트워크를 나눈 작은 단위.
* **모든 서브넷에 DHCP 서버를 따로 둘 필요 없이**, BOOTP 릴레이 에이전트를 통해 **다른 서브넷에서도 DHCP 요청을 처리할 수 있어야 함**.

**✅ 5. DHCP 클라이언트는 여러 개의 DHCP 서버 응답을 받을 준비가 되어 있어야 한다**

* **DHCP 서버가 여러 개 있을 수도 있음!**
* 클라이언트가 **하나 이상의 DHCP 서버로부터 응답을 받을 수 있어야 하고**, 적절한 설정을 선택할 수 있어야 함.

**✅ 6. DHCP는 기존의 수동 IP 설정 방식과 함께 동작할 수 있어야 한다**

* DHCP를 사용하지 않고 **수동으로 IP를 설정한 장치들과도 함께 작동해야 함**.
* DHCP가 기존 네트워크 프로토콜(BOOTP 포함)과 호환되어야 함.

**✅ 7. DHCP는 BOOTP 릴레이 에이전트와 호환되어야 한다**

* \*\*BOOTP 프로토콜(RFC 951, RFC 1542)\*\*에서 사용하는 릴레이 에이전트와 DHCP가 함께 작동할 수 있어야 함.
* BOOTP 클라이언트도 DHCP 서버로부터 설정을 받을 수 있어야 함.

**✅ 8. 네트워크 주소(IP)가 중복되지 않아야 한다**

* **같은 네트워크 주소(예: IP 주소)가 두 개의 장치에 동시에 할당되면 안 됨!**
* DHCP는 중복된 IP 주소를 방지해야 함.

**✅ 9. 클라이언트가 재부팅해도 설정이 유지되어야 한다**

* DHCP 클라이언트가 **재부팅해도 기존에 사용하던 IP 주소 및 설정을 그대로 유지해야 함**.
* 단, 네트워크 상황에 따라 다른 설정을 받을 수도 있음.

**✅ 10. DHCP 서버가 재부팅해도 기존 클라이언트 설정이 유지되어야 한다**

* DHCP 서버가 **재부팅되거나 장애가 발생해도 기존 클라이언트들의 설정이 유지되어야 함**.
* 즉, **서버가 다시 켜졌을 때 기존 클라이언트들이 같은 IP를 받을 수 있도록 해야 함**.

**✅ 11. 새로운 클라이언트도 자동으로 설정을 받을 수 있어야 한다**

* 네트워크에 **새로운 장치가 연결될 때도 DHCP가 자동으로 IP 주소 및 설정을 할당해야 함**.
* 관리자가 일일이 설정할 필요가 없어야 함.

**✅ 12. 특정 장치에 고정 IP 주소를 할당할 수도 있어야 한다**

* DHCP가 보통은 동적으로 IP를 할당하지만, **특정 장치(예: 서버, 프린터 등)에는 고정 IP를 할당할 수 있어야 함**.

# 2. Protocol Summary

**DHCP와 BOOTP의 관계**

DHCP는 기존 BOOTP 프로토콜을 확장한 형태입니다.

* 기존의 BOOTP 클라이언트는 변경 없이 DHCP 서버와 통신할 수 있습니다.
* BOOTP와 DHCP 간의 상호작용에 대한 자세한 내용은 RFC 1542에서 설명됩니다.
* DHCP는 BOOTP의 기본 기능을 유지하면서도, 추가적인 기능을 제공합니다.

**DHCP의 주요 차이점**

BOOTP와 비교했을 때, DHCP에는 두 가지 주요 차이점이 있습니다.

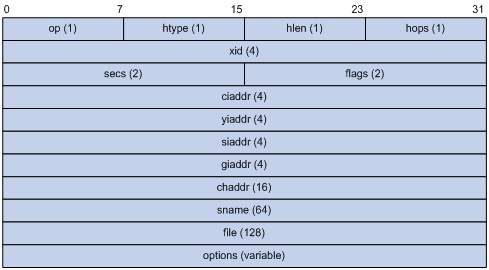
1. **임대(Lease) 개념 도입**
   * DHCP는 클라이언트가 **일정 기간 동안만** IP 주소를 사용할 수 있도록 합니다.
   * 즉, 네트워크 관리자는 동일한 IP 주소를 여러 장치에 순차적으로 할당할 수 있습니다.
2. **네트워크 설정 정보 제공**
   * DHCP는 단순히 IP 주소만 할당하는 것이 아니라,  
     **서브넷 마스크, 게이트웨이, DNS 서버 등의 추가 설정 정보**도 함께 제공합니다.

**BOOTP와 DHCP의 필드 변경**

* BOOTP에서 사용되던 **"vendor extensions" 필드**가 DHCP에서는 **"options" 필드**로 변경되었습니다.
* BOOTP의 "vendor extensions" 필드 내 데이터 항목들도 이제 \*\*"options"\*\*으로 불립니다.

**DHCP 메시지 구조**

아래는 DHCP 메시지의 기본 구조를 보여주는 표입니다.  
각 필드의 크기(단위: 바이트)와 역할을 설명하겠습니다.



텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**💡 참고:**

* options 필드는 DHCP의 핵심적인 기능을 담고 있으며, 다양한 설정 정보를 전달할 수 있습니다.
* DHCP 메시지는 최소 576 바이트의 크기를 가져야 하며, options 필드는 기본적으로 312 바이트 이상을 수용할 수 있어야 합니다.

**DHCP의 주요 개선 사항**

**1. 클라이언트 식별 방식 개선**

* DHCP는 **"client identifier"** 라는 새로운 옵션을 도입하여 클라이언트를 고유하게 식별합니다.
* 기존 BOOTP에서는 chaddr 필드가 클라이언트 식별자 역할을 했으나, DHCP에서는 별도의 식별자를 사용할 수 있습니다.
* 이 식별자는 MAC 주소일 수도 있고, DNS 이름과 같은 다른 식별자일 수도 있습니다.
* **중요:** 클라이언트가 한 번 client identifier를 설정하면 이후 모든 DHCP 메시지에서 동일한 값을 사용해야 합니다.

**2. siaddr 필드의 의미 명확화**

* DHCP에서는 siaddr 필드를 **다음 단계에서 사용할 서버의 IP 주소**로 정의합니다.
* 예를 들어, DHCP 서버가 운영체제 부팅 이미지를 제공하는 경우, siaddr 필드에 자신의 IP 주소를 포함시킬 수 있습니다.
* 또한, DHCP 서버는 항상 자신의 IP 주소를 "server identifier" 옵션으로 전달합니다.

**3. flags 필드 추가**

* DHCP는 flags 필드(2바이트)를 추가하여, 일부 클라이언트가 **유니캐스트**를 받을 수 없는 문제를 해결했습니다.
* flags 필드에서 가장 왼쪽 비트(Bit 0)는 **브로드캐스트(BROADCAST) 플래그**입니다.
* 이 비트가 1이면, 클라이언트는 서버가 **브로드캐스트 방식**으로 메시지를 보내길 원한다는 의미입니다.
* 나머지 비트들은 현재 사용되지 않으며, 항상 0으로 설정해야 합니다.  
  텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

  AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**DHCP 메시지 흐름 요약**

DHCP는 초기 설정 과정에서 TCP/IP 프로토콜을 완전히 구성하기 전에도 동작할 수 있도록 설계되었습니다.

* 클라이언트가 아직 IP 주소를 가지고 있지 않아도, DHCP 서버 또는 BOOTP 릴레이 에이전트가 메시지를 전달할 수 있도록 허용합니다.
* 일부 클라이언트는 TCP/IP가 완전히 설정되기 전에 유니캐스트 패킷을 수신할 수 없습니다.  
  → 이를 해결하기 위해 DHCP는 flags 필드를 활용하여 브로드캐스트 전송을 요청할 수 있습니다.

## 2.1 Configuration parameters repository

DHCP(동적 호스트 구성 프로토콜)의 첫 번째 기능은 네트워크 클라이언트의 구성 설정을 지속적으로 저장하는 것입니다. 이 저장소는 키-값 쌍으로 구성되어 있으며, 여기서 키는 고유한 식별자(예: IP 서브넷 번호와 해당 서브넷 내에서 고유한 식별자)이고, 값은 해당 클라이언트의 네트워크 구성 설정을 포함합니다.

예를 들어, 키는 (IP 서브넷 번호, 하드웨어 주소)로 구성될 수 있습니다. 이 "하드웨어 주소"는 장치의 물리적 주소를 의미하며, 다른 네트워크 구성이나 혼합된 네트워크에서 발생할 수 있는 충돌 문제를 고려해야 합니다. 이 방식은 동일한 하드웨어 주소가 서로 다른 서브넷에서 사용되거나, 전 세계적으로 고유하지 않은 하드웨어 주소가 발생할 수 있는 경우를 처리할 수 있게 해줍니다. 또는, 키가 (IP 서브넷 번호, 호스트 이름)일 수도 있습니다. 이는 클라이언트가 다른 서브넷으로 이동하거나 하드웨어 주소가 변경된 경우(예: 네트워크 인터페이스 고장으로 교체된 경우)에도 DHCP 서버가 지능적으로 구성 설정을 할당할 수 있도록 돕습니다.

기본적으로 사용되는 키는 (IP 서브넷 번호, 하드웨어 주소)이며, 클라이언트가 '클라이언트 식별자' 옵션을 사용해 고유 식별자를 제공하지 않는 한 이 키가 사용됩니다. 클라이언트는 DHCP 서비스에 쿼리하여 자신의 구성 설정을 조회할 수 있습니다. 클라이언트는 구성 매개변수 저장소와의 상호작용을 위해 구성 매개변수를 요청하는 프로토콜 메시지와 서버로부터의 응답을 통해 구성 설정을 받습니다.

## 2.2 Dynamic allocation of network addresses

DHCP의 두 번째 서비스는 클라이언트에게 임시 또는 영구적인 네트워크(IP) 주소를 할당하는 것입니다. 네트워크 주소의 동적 할당 기본 메커니즘은 간단합니다. 클라이언트는 일정 기간 동안 주소를 사용할 수 있도록 요청합니다. 할당 메커니즘(DHCP 서버의 모음)은 요청된 시간 동안 해당 주소를 재할당하지 않도록 보장하며, 클라이언트가 주소를 요청할 때마다 같은 네트워크 주소를 반환하려고 시도합니다. 이 문서에서는 네트워크 주소가 클라이언트에게 할당되는 기간을 "임대(lease)"라고 합니다. 클라이언트는 후속 요청을 통해 임대를 연장할 수 있습니다. 클라이언트가 더 이상 주소가 필요하지 않으면, 해당 주소를 서버로 반환하는 메시지를 전송할 수 있습니다. 또한, 클라이언트는 무한 임대를 요청하여 영구적인 할당을 받을 수 있습니다. "영구적인" 주소를 할당할 때도 서버는 주소를 영구적으로 할당하지 않고, 긴 기간의 임대를 제공하여 클라이언트가 더 이상 사용되지 않음을 감지할 수 있도록 할 수 있습니다.

어떤 환경에서는 사용할 수 있는 주소가 부족해져서 네트워크 주소를 재할당해야 할 필요가 있을 수 있습니다. 이런 환경에서는 할당 메커니즘이 임대가 만료된 주소를 재사용할 수 있습니다. 서버는 구성 정보 저장소에서 이용할 수 있는 정보를 사용하여 재사용할 주소를 선택해야 합니다. 예를 들어, 서버는 가장 최근에 할당된 주소를 선택하지 않을 수 있습니다. 일관성을 확인하기 위해, 할당 서버는 재사용할 주소를 할당하기 전에 해당 주소를 확인해야 하며, 예를 들어 ICMP 에코 요청을 사용하여 확인할 수 있습니다. 또한, 클라이언트는 새로 받은 주소에 대해 ARP와 같은 방법으로 확인을 해야 합니다.

# 3. The Client-Server Protocol - Todo

## 3.1 Client-server interaction - allocating a network address

## 3.2 Client-server interaction - reusing a previously allocated network address

## 3.3 Interpretation and representation of time values

## 3.4 Obtaining parameters with externally configured network address

## 3.5 Client parameters in DHCP

## 3.6 Use of DHCP in clients with multiple interfaces

## 3.7 When clients should use DHCP

# 4. Specification of the DHCP client-server protocol - Todo

## 4.1 Constructing and sending DHCP messages

## 4.2 DHCP server administrative controls

## 4.3 DHCP server behavior

### 4.3.1 DHCPDISCOVER message

### 4.3.2 DHCPREQUEST message

### 4.3.3 DHCPDECLINE message

### 4.3.4 DHCPRELEASE message

### 4.3.5 DHCPINFORM message

### 4.3.6 Client messages

## 4.4 DHCP client behavior

### 4.4.1 Initialization and allocation of network address

### 4.4.2 Initialization with known network address

### 4.4.3 Initialization with an externally assigned network address

### 4.4.4 Use of broadcast and unicast

### 4.4.5 Reacquisition and expiration

### 4.4.6 DHCPRELEASE

# 5. Acknowledgments

# 6. References

[1] RFC 887 "Resource Location Protocol"

[2] RFC 1533 "DHCP Options and BOOTP Vendor Extensions"

[3] RFC 1122 "Requirements for Internet Hosts -- Communication Layers"

[4] RFC 1123 "Requirements for Internet Hosts -- Application and Support"

[5] "Dynamic Reverse Address Resolution Protocol (DRARP)", Work in Progress

[6] "Uniform Access to Internet Directory Services"

[7] RFC 951 "Bootstrap Protocol (BOOTP)"

[8] RFC 1256 "ICMP Router Discovery Messages"

[9] RFC 1534 "Interoperation between DHCP and BOOTP"

[10] RFC 903 "A Reverse Address Resolution Protocol"

[11] "Leases: An Efficient Fault-Tolerant Mechanism for Distributed File Cache Consistency"

[12] RFC 1034 "Domain Names -- Concepts and Facilities"

[13] RFC 1035 "Domain Names -- Implementation and Specification"

[14] RFC 1191 "Path MTU Discovery"

[15] "Dynamic IP Address Assignment for Ethernet Attached Hosts"

[16] RFC 792 "Internet Control Message Protocol"

[17] RFC 1497 "BOOTP Vendor Information Extensions"

[18] RFC 1700 "Assigned Numbers"

[19] A Protocol for the Dynamic Assignment of IP Addresses for use on an Ethernet

[20] RFC 783 "The TFTP Protocol (Revision 2)"

[21] RFC 1542 "Clarifications and Extensions for the Bootstrap Protocol"

# 7. Security Considerations

# 8. Author's Address

# Appendix A. Host Configuration Parameters - Todo