# 8. Autoconfiguración de red y descubrimiento de servicios

Servicios Telemáticos Avanzados

4º Grado en Ingeniería en Tecnologías de Telecomunicación

Especialidad de Telemática

# Indice

*Hora 1*

1. Introducción
2. Asignación automática de parámetros de red con presencia de servidores
3. Zeroconf
3.1 Asignación de direcciones IP: escenarios y requerimientos
3.1.1 Conflict-detection allocation
3.1.2 Conflict-free allocation
3.1.3 Best effort allocation

*Hora 2*

3.2 Traducción de nombres
3.3 Asignación de direcciones IP multicast
3.4 Descubrimiento de servicios
4. Universal Plug and Play
5. Jini

# 1. Introducción

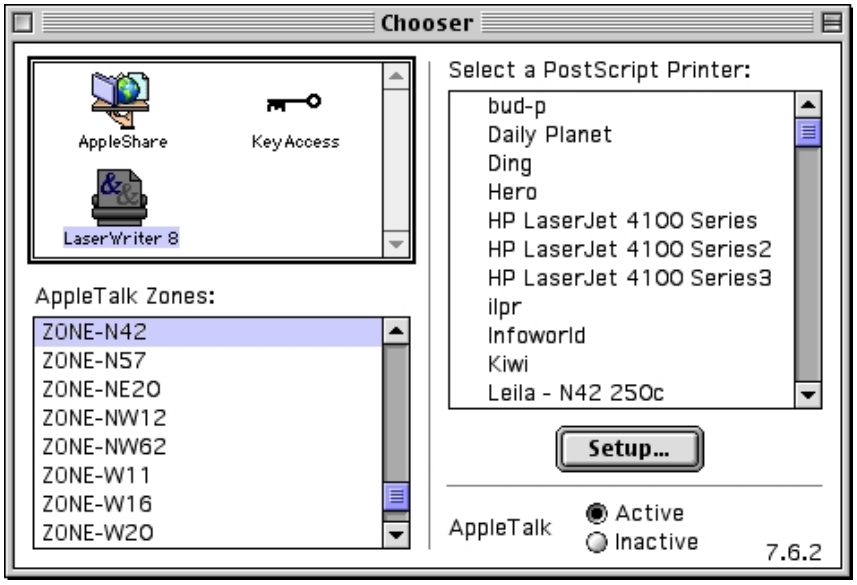
- Uno de los grandes problemas que ha tenido tradicionalmente el mundo IP ha sido el de la configuración del direccionamiento de los equipos y el acceso a los servicios:
  - Direccionamiento: sistemas de asignación manual o basado en servidores (DHCP)
  - Acceso a servicios: conocimiento a priori por parte del cliente
- Si se quiere facilitar su uso, con la menor intervención posible del usuario, es necesario
  - Poder manejar la configuración de direccionamiento y otras configuraciones de los equipos de forma transparente para el usuario sin necesidad de servidores DHCP
  - Traducción entre nombres y direcciones sin necesidad de servidores de DNS
  - Descubrimiento de servicios automático, sin necesidad de disponer de un conocimiento a priori

# Introducción

- Funcionalidades a automatizar
  - Configuración de red
    - Se trata de evitar solicitar al usuario una configuración de red cuando las funcionalidades que necesita son mínimas, por ejemplo compartir ficheros entre máquinas de la misma LAN.
  - Traducción de nombres
    - Mapeo automático entre direcciones IP y nombres sin necesidad de servidores de DNS.
  - Localización de servicios
    - Encontrar servicios sin conocer a priori su ubicación (dirección IP, puerto, descripción del servicio).

# Introducción

- Estas funcionalidades sí han sido habituales en otros sistemas no IP:
  - AppleTalk, desde las primeras versiones del sistemas Apple Mac
    - Una de sus grandes características desde sus inicios
  - Microsoft NETBIOS, en sistemas operativos Windows
  - Novell IPX, en sistemas operativos Windows
    - Muy valorado en pequeñas redes empresariales



# 2. Asignación automática de parámetros de red con presencia de servidores

- Protocolos que proveen asignación dinámica de información de red:
  - RARP, Reverse Address Resolution Protocol.
    - Permite el proceso inverso al ARP de obtención de la dirección IP de la máquina conociendo su dirección MAC.
    - Información de red que proporciona es insuficiente.
  - BOOTP, Bootstrap Protocol
    - Sobre UDP, puertos 67(servidor)-68(cliente).
    - Solicitud sobre paquete con IP destino broadcast.
    - Permite especificar la máscara, router por defecto, DNS e incluso un fichero con código de arranque para la máquina.
    - Mapeo fijo IP-MAC en el servidor.
  - DHCP, Dynamic Host Configuration Protocol
    - Como BOOTP pero permite la asignación de direcciones IP temporales o no asignadas a priori a una dirección MAC.

Siguientes figuras:

- xid: ID transacción
- ciaddr: dirección IP cliente
- yiaddr: dirección IP asignada
- serverID: dirección IP servidor

# DHCP solicitud inicial

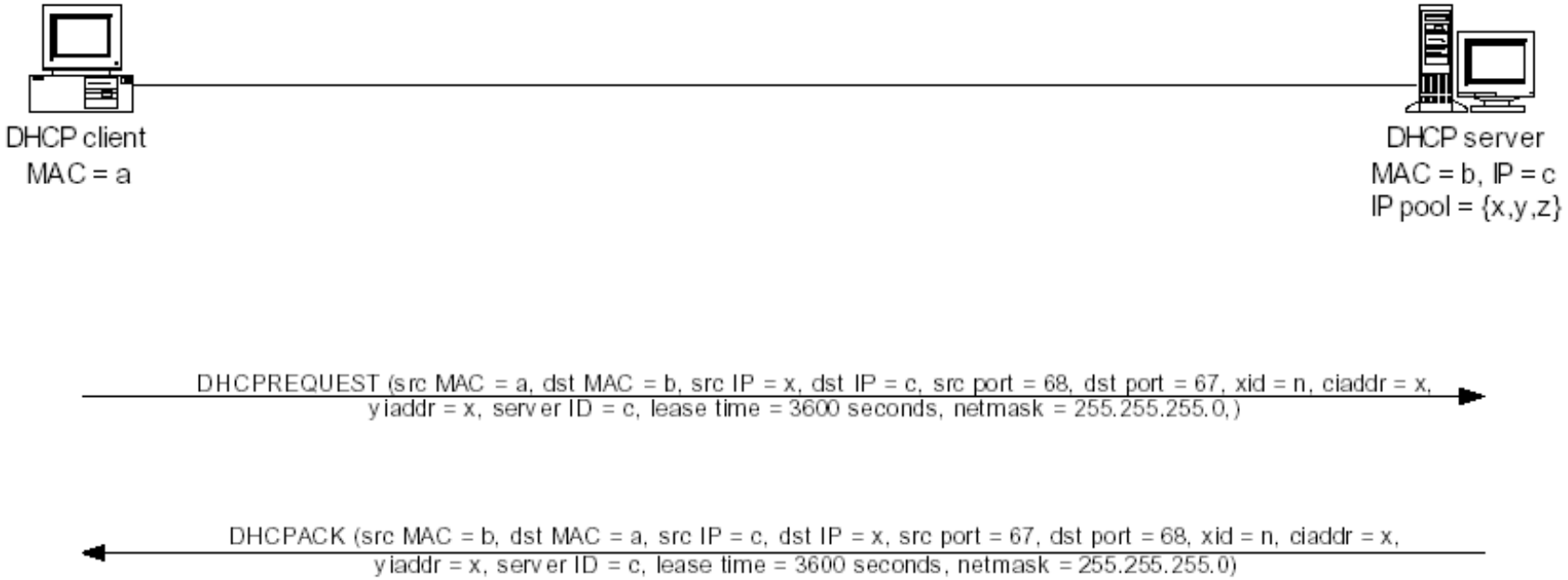

DHCPDISCOVER (src MAC = a, dst MAC = 0xffffffffffff, src IP = 0.0.0.0, dst IP = 255.255.255.255, src port = 68, dst port = 67, xid = n, ciaddr = 0.0.0.0) →

← DHCPOFFER (src MAC = b, dst MAC = 0xffffffffffff, src IP = c, dst IP = 255.255.255.255, src port = 67, dst port = 68, xid = n, ciaddr = 0.0.0.0, yiaddr = x, server ID = c, lease time = 3600 seconds, netmask = 255.255.255.0)

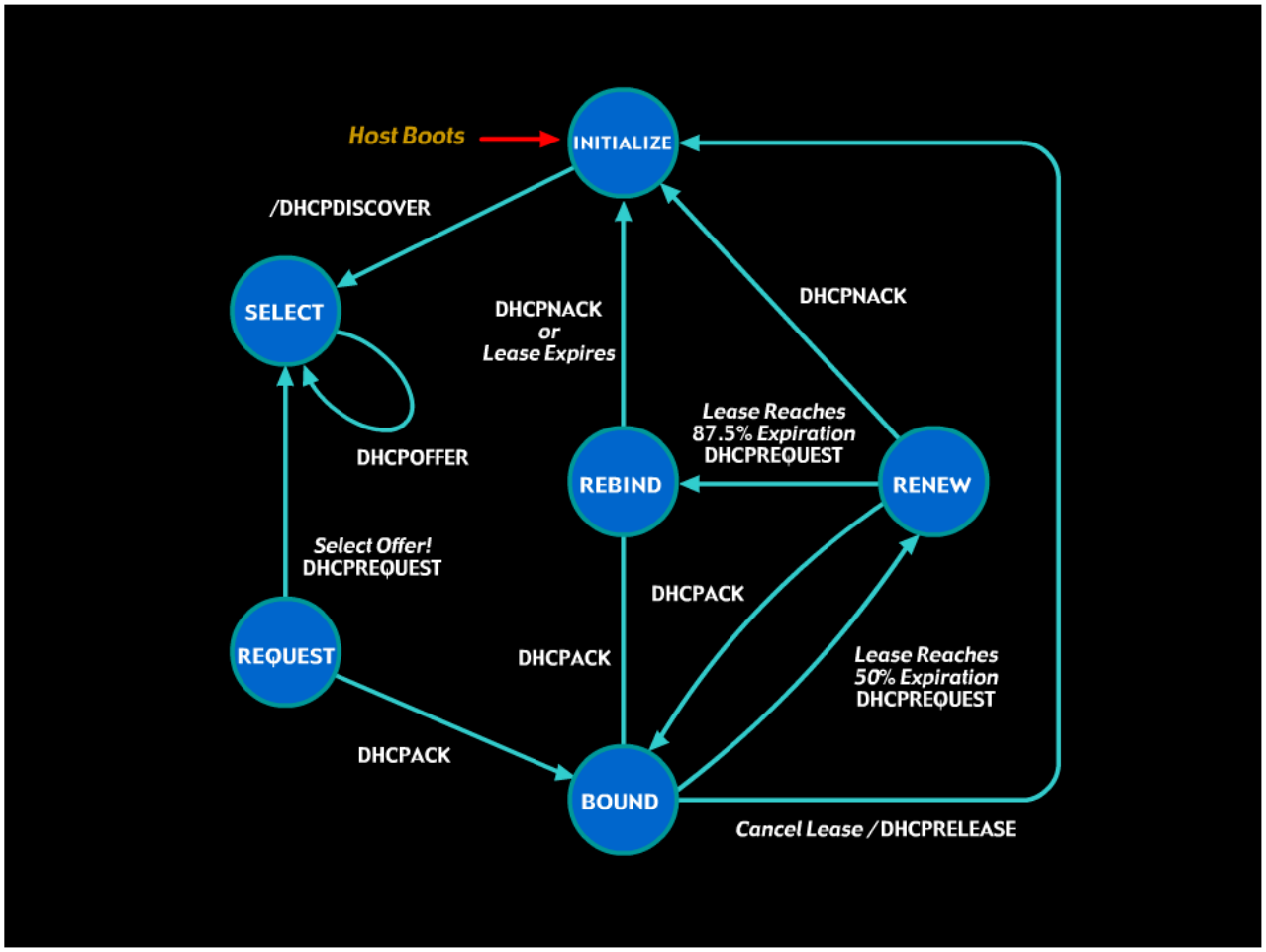
DHCPREQUEST (src MAC = a, dst MAC = 0xffffffffffff, src IP = 0.0.0.0, dst IP = 255.255.255.255, src port = 68, dst port = 67, xid = n+1, ciaddr = 0.0.0.0, yiaddr = x, server ID = c, lease time = 3600 seconds, netmask = 255.255.255.0, requested IP address = x) →

← DHCPACK (src MAC = b, dst MAC = 0xffffffffffff, src IP = c, dst IP = 255.255.255.255, src port = 67, dst port = 68, xid = n+1, ciaddr = 0.0.0.0, yiaddr = x, server ID = c, lease time = 3600 seconds, netmask = 255.255.255.0)

# DHCP renovación



# DHCP estados cliente



# DHCP opciones

**Table 8-1** DHCP Options

| Tag | Name | Length | Meaning |
|---|---|---|---|
| 0 | Pad | 0 | None |
| 1 | Subnet Mask | 4 | Subnet mask value |
| 2 | Time Offset | 4 | Time offset in seconds from UTC |
| 3 | Router | N | N/4 router addresses |
| 4 | Time Server | N | N/4 time server addresses |
| 5 | Name Server | N | N/4 IEN-116 server addresses |
| 6 | Domain Server | N | N/4 DNS server addresses |
| 7 | Log Server | N | N/4 logging server addresses |
| 8 | Quotes Server | N | N/4 quotes server addresses |
| 9 | LPR Server | N | N/4 printer server addresses |
| 10 | Impress Server | N | N/4 Impress server addresses |
| 11 | RLP Server | N | N/4 RLP server addresses |
| 12 | Hostname | N | Hostname string |
| 13 | Boot File Size | 2 | Size of boot file in 512-byte chunks |
| 14 | Merit Dump File | N | Client to dump and name the file to dump it to |
| 15 | Domain Name | N | DNS domain name of the client |
| 16 | Swap Server | N | Swap server address |
| 17 | Root Path | N | Path name for root disk |
| 18 | Extension File | N | Path name for more BOOTP info |
| 19 | Forward On/Off | 1 | Enable/disable IP forwarding |
| 20 | SrcRte On/Off | 1 | Enable/disable source routing |
| 21 | Policy Filter | N | Routing policy filters |
| 22 | Max DG Assembly | 2 | Max datagram reassembly size |
| 23 | Default IP TTL | 1 | Default IP Time to Live |
| 24 | MTU Timeout | 4 | Path MTU aging timeout |
| 25 | MTU Plateau | N | Path MTU plateau table |
| 26 | MTU Interface | 2 | Interface MTU size |
| 27 | MTU Subnet | 1 | All subnets are local |
| 28 | Broadcast Address | 4 | Broadcast address |
| 29 | Mask Discovery | 1 | Perform mask discovery |
| 30 | Mask Supplier | 1 | Provide mask to others |
| 31 | Router Discovery | 1 | Perform router discovery |
| 32 | Router Request | 4 | Router solicitation address |
| 33 | Static Route | N | Static routing table |
| 34 | Trailers | 1 | Trailer encapsulation |

**Table 8-1** DHCP Options (continued)

| Tag | Name | Length | Meaning |
|---|---|---|---|
| 35 | ARP Timeout | 4 | ARP cache timeout |
| 36 | Ethernet | 1 | Ethernet encapsulation |
| 37 | Default TCP TTL | 1 | Default TCP Time to Live |
| 38 | Keepalive Time | 4 | TCP keep-alive interval |
| 39 | Keepalive Data | 1 | TCP keep-alive garbage |
| 40 | NIS Domain | N | NIS domain name |
| 41 | NIS Servers | N | NIS server addresses |
| 42 | NTP Servers | N | NTP server addresses |
| 43 | Vendor Specific | N | Vendor-specific information |
| 44 | NETBIOS Name Srv | N | NETBIOS name servers |
| 45 | NETBIOS Dist Srv | N | NETBIOS datagram distribution |
| 46 | NETBIOS Node Type | 1 | NETBIOS node type |
| 47 | NETBIOS Scope | N | NETBIOS scope |
| 48 | X Window Font | N | X Window font server |
| 49 | X Window Manager | N | X Window display manager |
| 50 | Address Request | 4 | Requested IP address |
| 51 | Address Time | 4 | IP address lease time |
| 52 | Overload | 1 | Overload "sname" or "file" |
| 53 | DHCP Msg Type | 1 | DHCP message type |
| 54 | DHCP Server Id | 4 | DHCP server identification |
| 55 | Parameter List | N | Parameter request list |
| 56 | DHCP Message | N | DHCP error message |
| 57 | DHCP Max Msg Size | 2 | DHCP maximum message size |
| 58 | Renewal Time | 4 | DHCP renewal time (T1) |
| 59 | Rebinding Time | 4 | DHCP rebinding time (T2) |
| 60 | Vendor Class ID | N | Vendor class Identifier |
| 61 | Client ID | N | Client Identifier |
| 62 | NetWare/IP Domain | N | NetWare/IP domain name |
| 63 | NetWare/IP Option | N | NetWare/IP sub options |
| 64 | NIS-Domain-Name | N | NIS+ v3 client domain name |
| 65 | NIS-Server-Addr | N | NIS+ v3 server addresses |
| 66 | Server-Name | N | TFTP server name |
| 67 | Bootfile-Name | N | Boot filename |
| 68 | Home-Agent-Addrs | N | Home agent addresses |
| 69 | SMTP-Server | N | Simple Mail server addresses |

**Table 8-1** DHCP Options (continued)

| Tag | Name | Length | Meaning |
|---|---|---|---|
| 70 | POP3-Server | N | Post Office server addresses |
| 71 | NNTP-Server | N | Network News server addresses |
| 72 | WWW-Server | N | WWW server addresses |
| 73 | Finger-Server | N | Finger server addresses |
| 74 | IRC-Server | N | Chat server addresses |
| 75 | StreetTalk-Server | N | StreetTalk server addresses |
| 76 | STDA-Server | N | ST Directory Assistance addresses |
| 77 | User-Class | N | User class information |
| 78 | Directory Agent | N | Directory agent information |
| 79 | Service Scope | N | Service location agent scope |
| 80 | Naming Authority | N | Naming authority |
| 81 | Client FQDN | N | Fully qualified domain name |
| 82 | Agent Circuit ID | N | Agent circuit ID |
| 83 | Agent Remote ID | N | Agent remote ID |
| 84 | Agent Subnet Mask | N | Agent subnet mask |
| 85 | NDS Servers | N | Novell Directory Services |
| 86 | NDS Tree Name | N | Novell Directory Services |
| 87 | NDS Context | N | Novell Directory Services |
| 88 | IEEE 1003.1 POSIX | N | IEEE 1003.1 POSIX time zone |
| 89 | FQDN | N | Fully qualified domain name |
| 90 | Authentication | N | Authentication |
| 91 | Vines TCP/IP | N | Vines TCP/IP server option |
| 92 | Server Selection | N | Server selection option |
| 93 | Client System | N | Client system architecture |
| 94 | Client NDI | N | Client network device interface |
| 95 | LDAP | N | Lightweight Directory Access Protocol |
| 96 | IPv6 Transitions | N | IPv6 transitions |
| 97 | UUID/GUID | N | UUID/GUID-based Client Identifier |
| 98 | User-Auth | N | Open Group's user authentication |

"N" in length column represents a variable number.

# 3. Zeroconf

- Zero Configuration Networking
  - Grupo de trabajo del *IETF*
  - Establecido en septiembre de 1999
  - http://www.zeroconf.org/
- Objetivo mínimo: dos ordenadores conectados entre sí mediante un cable cruzado por Ethernet o en la misma LAN , han de comunicarse entre sí bajo IP, sin necesidad de intervención humana, ni servidores DHCP o DNS
- Implementaciones más exitosas
  - Bonjour de Apple, anteriormente llamado Rendezvous, estándar de los sistemas operativos MacOSX desde 2002.
  - http://www.apple.com/support/bonjour/

# Zeroconf: áreas de trabajo

- Para lograr esta funcionalidad con IP en pequeñas redes se crean cuatro áreas de trabajo principales:
  - Asignar direcciones IP sin un servidor DHCP (con dirección de red, router...)
  - Traducir entre nombres y direcciones IP sin un servidor DNS
  - Asignar direcciones IP multicast sin un servidor MADCAP (Multicast Address Dynamic Client Allocation Protocol)
  - Descubrir servicios, como por ejemplo impresoras, sin un Servicio de Directorio
- Las soluciones en cualquiera de las cuatro áreas han de coexistir amigablemente con las redes actualmente configuradas
  - Direccionamiento IP tanto de IPv4 como de IPv6
- Los protocolos de Zeroconf no tienen que causar perjuicio alguno a la red, cuando una máquina configurada con Zeroconf sea conectada a la red actual
  - Coexistencia con otras configuraciones manuales o vía servidor
  - Características de seguridad suficientes para prevenir que no sean menos seguros

# Zeroconf: objetivos

- Las funciones habrán de ser definidas para dos topologías de red distintas
  - Un segmento de red simple, donde los hosts son accesibles a través de la capa de enlace mediante broadcasting o mensajes multicast
  - Un conjunto de segmentos de redes (en distintas subredes IP) interconectadas mediante un simple router
    - La configuración automática de una topología arbitraria de routers y subredes queda fuera del ámbito del grupo de trabajo
- Definirá cómo una red puede automáticamente realizar una transición desde el comportamiento de configurada a desconfigurada y viceversa
  - Los mismos hosts han de ser capaces de funcionar en redes sin configuración, así como con conectividad directa IP hacia Internet, incluyendo servicios DNS, etc.
  - También será posible que ambos modos (Zeroconf y administrado) puedan coexistir en la misma red, sin ser dichos modos mutuamente excluyentes
- Simplicidad y facilidad de uso.

# Zeroconf: consideraciones de seguridad

- El principal avance de los protocolos Zeroconf es proveer configuración de la red, allá donde los servicios de configuración no están disponibles. Esto es ventajoso con operaciones seguras, pues los mecanismos de seguridad requieren generalmente alguna preconfiguración (claves, certificados, etc.)
- Normalmente, los mecanismos de seguridad en protocolos *IETF* son de implementación obligatoria, aunque una implementación particular quizá puede permitir a un administrador desactivar operacionalmente un mecanismo de seguridad. En cualquier caso, las implementaciones han de ser “seguras fuera de la caja” y han de configurarse seguras por defecto
- Los protocolos Zeroconf no pueden ser menos seguros que los protocolos actualmente relacionados del *IETF* estándar
- Las amenazas a considerar incluyen ataques activos (v.g. denegación de servicio) como ataques pasivos (escuchas a través de la red), y los protocolos que requieren confidencialidad y/o integridad deben resolverse mediante integración o usando los mecanismos estándar de seguridad

# 3.1 Asignación de direcciones IP: escenarios y requerimientos

- Configuración de la interfaz IP:
  - Siempre incluye la configuración de una dirección IP y de la máscara de red
  - Puede incluir alguna información de encaminamiento (p.ej., router por defecto)
  - Es necesario disponer de ella antes que ninguna comunicación se lleve a cabo
- Requerimientos:
  - Ha de configurar una máscara de red apropiada
  - Ha de tener una dirección IP única dentro de una subred
  - Ha de tener alguna información relativa al encaminamiento para la inter-red
  - Ha de tener una subred IP única dentro de la inter-red, si ésta existe
  - Tiene que resolver conflictos puntualmente ante los cambios de topología

# Asignación de direcciones IP: estrategias

- Existen tres estrategias principales:
  - Conflict-detection allocation
    - En este esquema los nodos conjeturan una IP (aleatoriamente o utilizando alguna información de la red) y luego deben utilizar algún método para detectar direcciones duplicadas.
  - Conflict-free allocation
    - En este esquema no hay conflicto de direcciones, ya que se implementan mecanismos que controlan a priori que no pueda existir un solapamiento de las direcciones. Se basan en algoritmos de asignación de enteros para que los conjuntos sean disjuntos.
  - Best-effort allocation
    - Los nodos responsables de la asignación de direcciones intentan asignar direcciones IP no utilizadas en la medida de la información de que disponen y luego utilizan técnicas de detección de conflictos para los casos en los que haya conflicto.

# 3.1.1 Conflict-detection allocation

- Automatic Private IP Addressing (APIPA) es la que utliza Zeroconf
- Se utilizan direcciones de clase B con prefijo 169.254.0.0/16 IPv4 y FE80::/10 en IPv6.
- El nodo que intenta obtener una dirección IP utiliza inicialmente una temporal (para comunicarse con el resto) dentro del rango 0-2047, seleccionándola aleatoriamente.
- Elige otra, dentro del rango 2048-65534, como dirección tentativa.

# Conflict-detection allocation

- El nodo envía un Address Request (AREQ) por broadcast a sus vecinos y arranca un temporizador.
  - El AREQ contiene la dirección temporal y la tentativa.
- Cuando un nodo recibe un AREQ comprueba que no coincida con su IP la dirección tentativa.
  - Si no coincide reenvía el mensaje a sus vecinos (broadcast) lo mismo hace con los AREP que no son para él.
  - Si coincide envía (por broadcast) un Address Reply (AREP).
- Cuando un nodo envía AREQ espera AREP hasta que venza el time-out. Si es así repite el proceso AREQ_TIMES
  - si no hay contestación se queda la IP, considerándose configurado.
  - si el nodo recibe AREP, vuelve a iniciar el proceso con otra dirección.

# 3.1.2 Conflict-free allocation

- Esquemas heredados de redes MANET (Mobile Ad-Hoc Network)
- Enfocan el problema de la autoconfiguración de direcciones IP como el de la asignación de un conjunto de enteros dentro de un rango dado
- Convenientemente realizada la asignación, no tiene porqué haber conflictos.
  - El nodo inicial sabe a priori que conflictos se pueden producir, pudiendo evitarlos realizando la detección antes de la asignación de direcciones

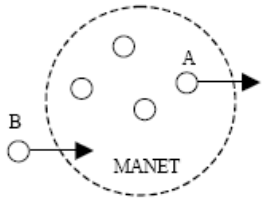



Figure 1. A node joins and leaves the MANET once

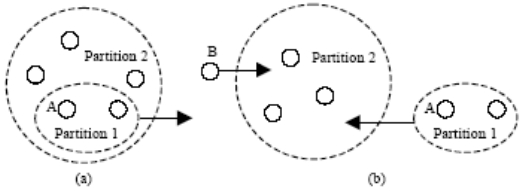



Figure 2. Network partitions and merges

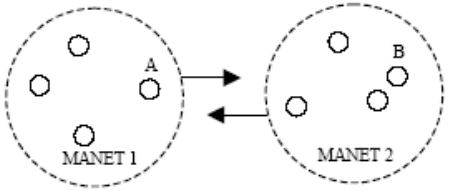
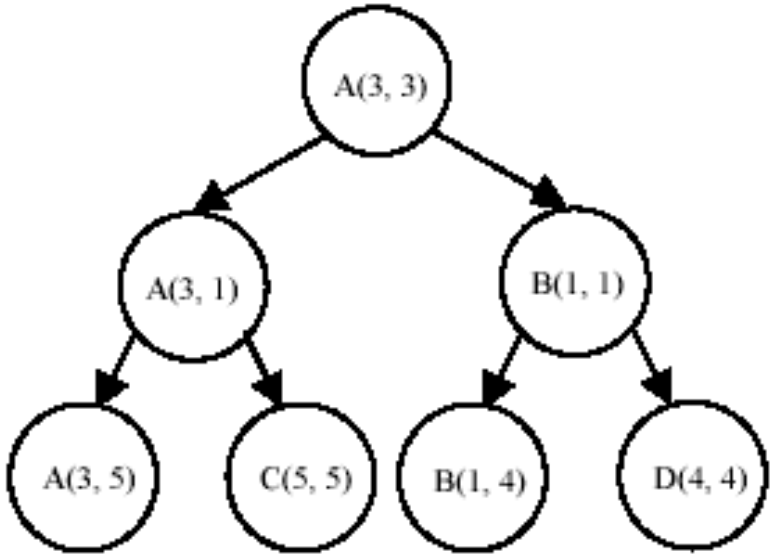



Figure 3. Merger of two indpendent MANETS

# Conflict-free allocation

- Ejemplo de funcionamiento:
  - R∈[1,8]
  - f(n)=(address x semilla x 11) mod 7
  - cada nodo tiene asociado: (address, semilla f(n)).




- Cuando la red se inicia sólo está A. Elige aleatoriamente el número 3 como dirección IP y como semilla para f(n).
- Cuando B intenta unirse, A calcula f(3)=(3 x 3 x11) mod 7=1 y se lo pasa a B como dirección IP y semilla. Además actualiza su semilla con ese valor.
- Posteriormente C se aproxima a A y D a B. Cada uno calcula los valores independientemente (pero compartiendo semilla) de forma que
  - para C: f(n)=f(1)=(3 x 1 x 11) mod 7=5 → IP C=5 y el estado de A y C=5.
  - para B: f(n)=f(1)=( 1 x 1 x 11) mod 7=4 → IP D=4 y el estado de B y D=4.

# 3.1.3 Best effort allocation

- Combinación de los dos anteriores
- Protocolo distribuido, no impide el conflicto de direcciones pero garantiza la no duplicación a costa de:
  - Mantener mucha información de estado por nodo
  - Aplicación de DAD (Duplicate Address Detection)
- Permite
  - Guardar menos estado que los conflict-free
  - Escalar en redes con mayor número de máquinas