Gdansk University of Technology

Practical aspects of DHCPv6 deployment

Tomasz Mrugalski

October 20th, 2008
Stockholm

Agenda

- Introduction
- DHCPv6 overview
- Usage scenarios
- Dibbler project
- DHCPv6 interop events
- Q&A

Introduction

About author

Tomasz Mrugalski

- M. Sc. at Gdansk University of Technology in 2003
- Doing Ph.D related to IPv6/WiMAX/DHCPv6
- Working as network engineer at Intel Corp. since 2004
- Developing open-source DHCPv6 implementation since 2003
- Participated in 3 DHCPv6 interoperability events
- Several publications regarding DHCPv6:
  - 5 IEEE conferences (Poland, Croatia, New Zealand, Australia)
- Polish IPv6 Task Force member since 2005

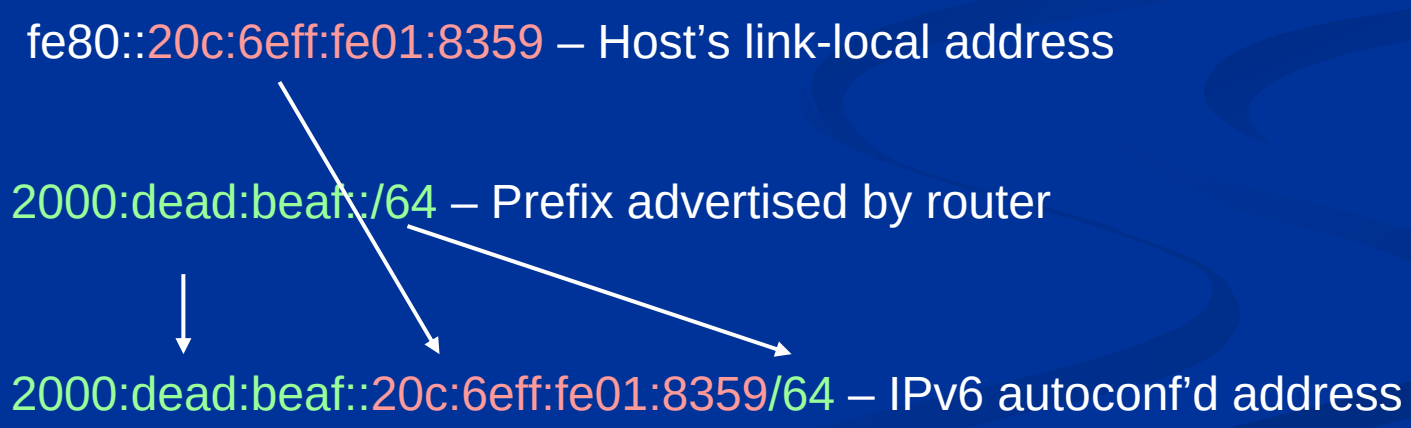
DHCPv6 overview

IPv6 autoconfiguration

- Stateless autoconfiguration
  - Link-local addresses
  - Global addresses (Router Advertisements)
  - Lack of per host customization
  - Does not support any additional parameters
  - Not sufficient for end users (DNS)
- Stateful autoconfiguration (DHCPv6)
  - ***Additional parameters*** configuration (e.g. **DNS**)
  - Global address assignment
  - Does not configure routing
  - Stateful = state of every node is known and monitored
  - Provides network status for administrators

Stateless autoconfiguration :: Router Advertisements

- RAs sent by routers
  - Prefixes available through this router
    - Nodes configure routing (via router)
  - Prefixes available locally on the link
    - Nodes configure routing (directly)
  - Autonomous prefixes
    - Nodes configure routing (directly)
    - Nodes obtain new addresses



DHCPv6 :: Overview (1)

- Specified in 2003 by IETF [RFC3315]
  - Largest number of draft revisions ever
  - 20+ RFCs related to DHCPv6
  - Numerous drafts
- Consists of clients, servers, relays and requestors (leasequery)
- Framework for server redundancy and failover
- Multicast-based (mcast is supported by all IPv6 nodes)
  - Optional unicast support
- Supports Duplicate Address Detection
- Generic mechanism for configuring IPv6 addresses and more:
  - Delegating prefixes
  - FQDN allocations, DNS updates
  - 20+ options (SIP servers, domains, DNS, timezones, subscriber-ids, …)
  - Options renewal
  - Lease queries
  - Temporary addresses
  - Relays (including cascade)
  - Unicast communication
  - …

DHCPv6 :: Overview (2)

- Negotiation mechanisms available
- Options
  - DNS servers configuration
  - Domain configuration
  - NTP servers configuration
  - Timezone configuration
  - SIP-related parameters configuration
  - NIS, NIS+ parameters configuration
  - Boot options
  - Vendor-specific information
  - Subscriber-ID, Remote-ID
  - Authentication
  - Echo Request
  - …
- Authentication
- Reconfigure support

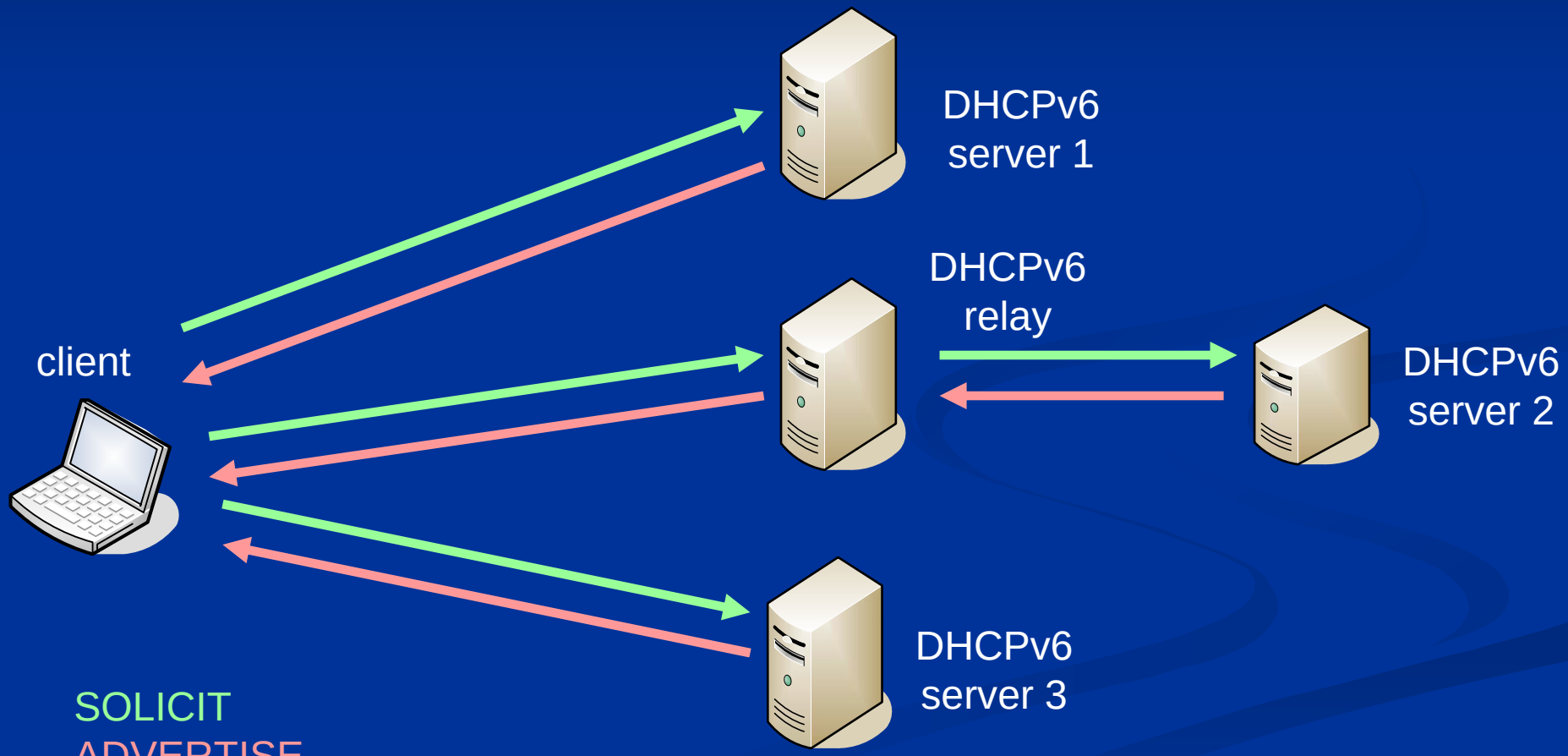
Usage scenarios

DHCPv6 :: Basic address acquisition

- Client sends SOLICIT
  - Contains desired parameters (addresses and options)
  - „Hey, who can assign me these?”
- All servers available respond with ADVERTISE
  - Contains possible server responses
  - Contains server identifier
  - „If you choose me, I will grant you something like this”
- Client sends REQUEST to a chosen server
  - „Assign me those options, please”
- Server grants addresses and options and sends REPLY
  - „These are your parameters”

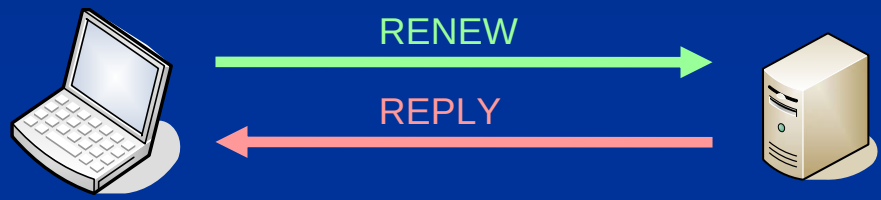
DHCPv6 :: Server discovery

- Client sends SOLICIT to locate servers and its capabilities
- All servers available respond
- Client chooses best server and asks for parameters



DHCPv6 :: Renewal mechanism

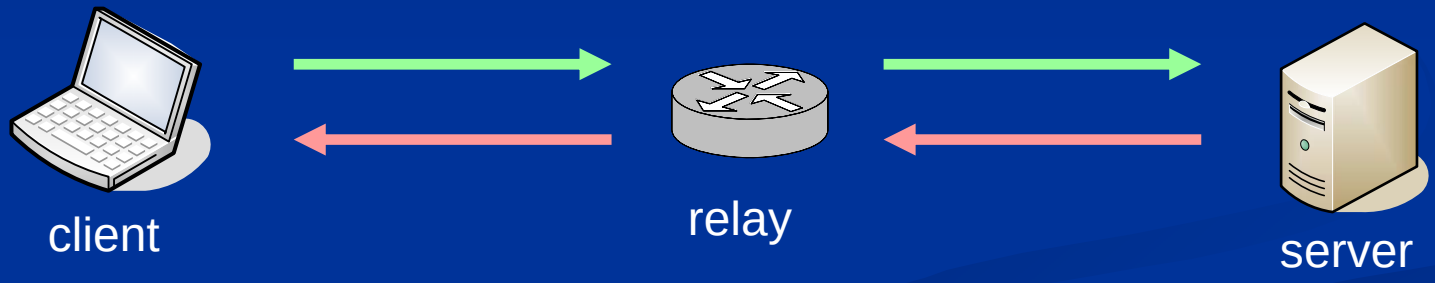
- Renewal mechanism
  - RENEW contains server identifier and is addresses to a specific server
  - RENEW is ignored by other servers
  - Asked SERVER responds with updated parameters (e.g. extented address lifetimes)



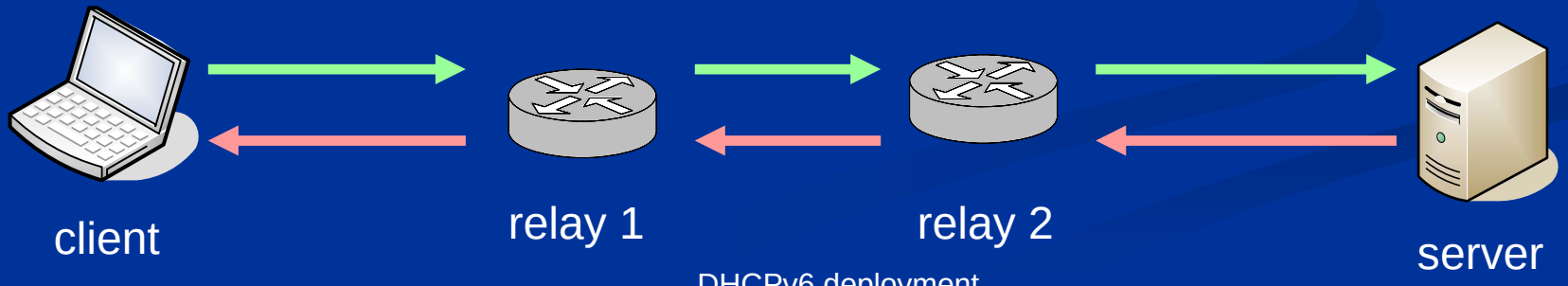
- Emergency Renewal mechanism
  - Used when after several retransmissions RENEW remains unanswered)
  - Does not contain server identifier („addressed to any server”)
  - One of the servers (e.g. server2) available respond
  - From now on, client will renew its parameters from server2

DHCPv6 :: Relays

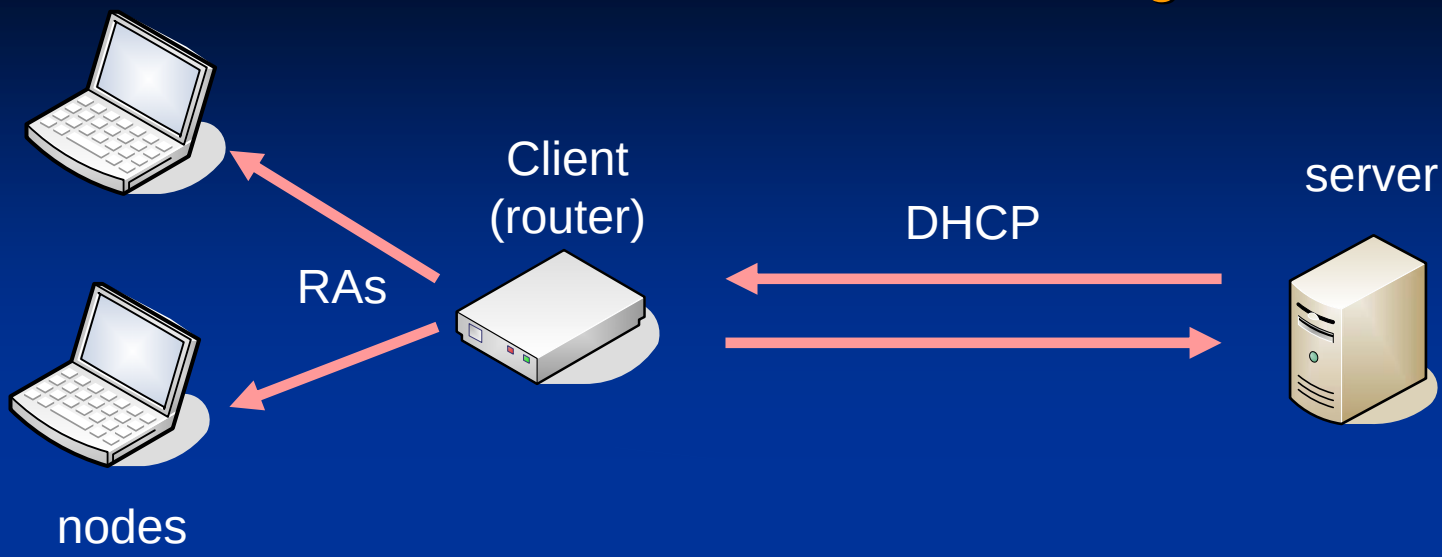
- Relay support
  - Server supports multiple links (local and remote)
  - Relays can add extra options:
    - Remote-ID
    - Subscriber-ID
    - Echo option



- Cascade relays support
  - Up to 32 relays
  - Great scalability

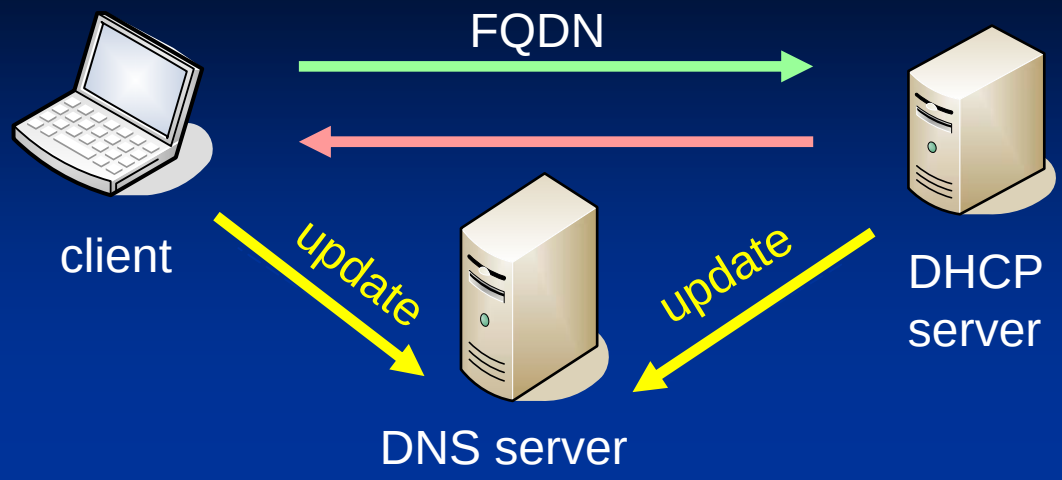


DHCPv6 :: Prefix delegation



- Client is not the end-user, but a CPE
  - Receives whole prefix intead of single address
  - Prefix can be announced over user's link
  - Can be also used for local mgmt (e.g. serving addresses via DHCPv6 to users)

DHCPv6 :: DNS Update

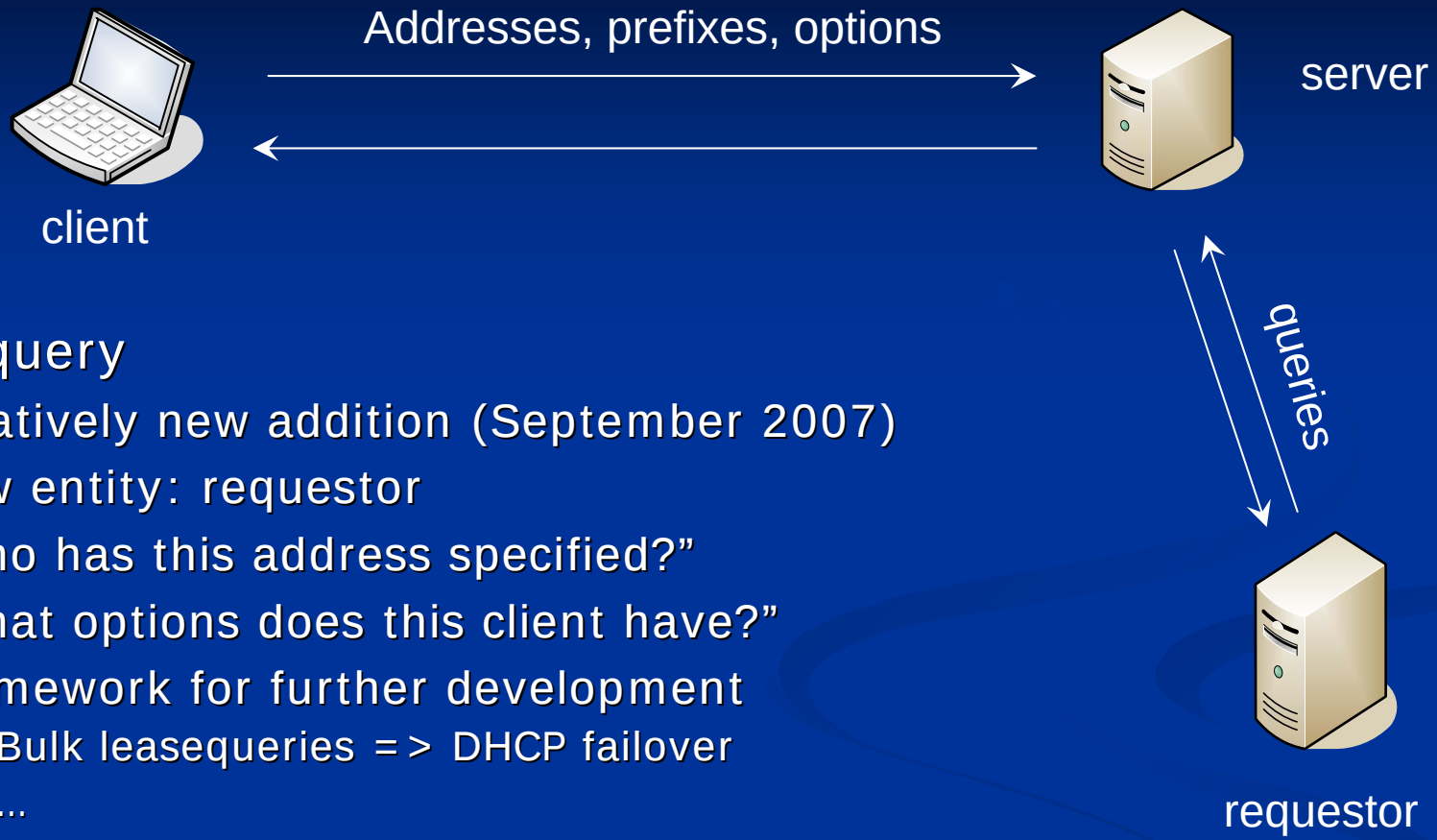


- Fully Qualified Domain Name
  - Client wants to obtain fully qualified domain name
  - Client wants to obtain hostname
  - ...
- Several scenarios
  - Forward resolution only (AAAA records)
  - Reverse resolution only (PTR records)
  - Both
  - Performed by server (better security, worse scalability)
  - Performed by client (better scalability, worse security)

DHCPv6 :: Address duplicates

- Client checks address uniqueness
  - uses DAD (Duplicate Address Detection)
- Address can be rejected by client
- Server marks address as “dirty”

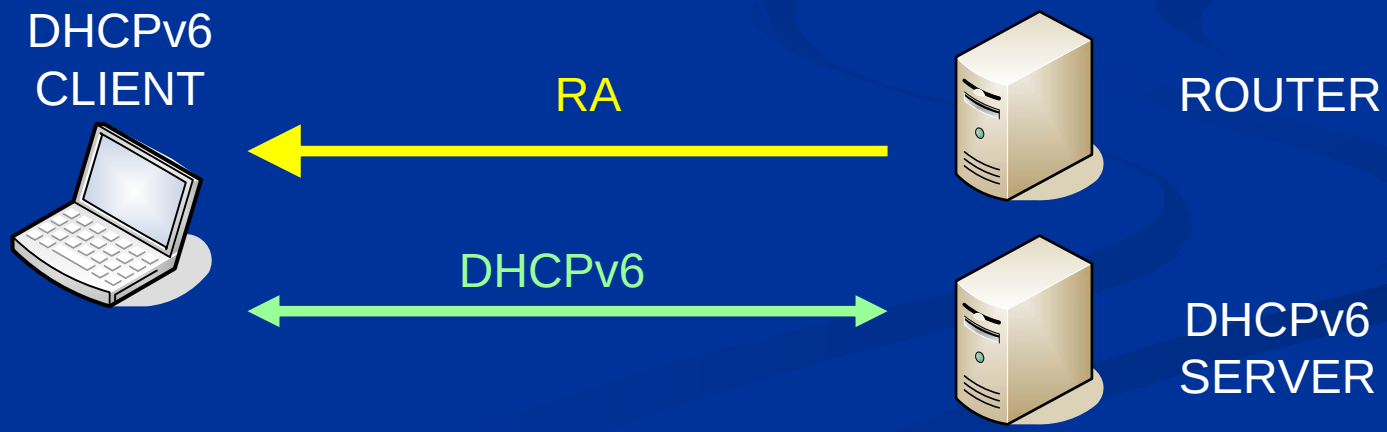
DHCPv6 :: Leasequery



- Leasequery
  - Relatively new addition (September 2007)
  - New entity: requestor
  - „Who has this address specified?”
  - „What options does this client have?”
  - Framework for further development
    - Bulk leasequeries => DHCP failover
    - ...
  - Work in progress

DHCPv6 / RA cooperation

- Client receives RA
  - M (managed) bit set => start DHCPv6 for managed (e.g. address)
  - O (other) bit set => ask for other options in DHCPv6
  - Routing for local prefix configured
  - Default gateway set
- Client starts DHCPv6
  - Obtains an addresses, prefixes (+extra options)



Dibbler project

The Dibbler project is an open source, portable implementation of the DHCPv6 protocol, providing client, server, relay and requestor. Currently ports for Linux and Windows based systems are available.

Dibbler :: Design assumptions (1)

- Extensive server configuration
  - Preferences, multiple classes on one interface, address-client reservation, client classification, multiple pools, temporary addresses, prefix delegation, white-list, black-list, numerous options etc.
- Zero client konfiguration
  - out-of-the-box (quick install)
- Non-mandatory client configuration
  - For experienced/more demanding users
- Portability
  - 2 layers: logic (C++), low-level functions (C)
  - Currently: Windows NT4/2k/XP/2k3, Linux 2.4/2.6
  - Porting to other systems requires reimplementation of small number of low level functions

Dibbler :: Design assumptions (2)

- Clean architecture
  - Object-oriented
  - Easily extended
- Extensibility
  - Additonal options may be defined
  - Easy services convergence
    - Data is dumped in XML (scripts,stats)
- Open source
  - Free for use, including commercial use
  - Sources available
  - GNU GPL licence

Dibbler :: Current Status (1)

- Started in 2003 as double master thesis
  - Grown considerably since then...
- Feedback from 29 countries
  - Poland, Germany, Chech Republic, France, Spain, USA, China, Malaysia, Canada, Taiwan, Switzerland, Turkey, India, UK, Austria, Hungary, Cuba, Japan, Sweden, Luxembourg, Australia, Israel, Norway, Thailand, Finland, Philippines, Venezuela, Bosna and Herzegovina, Portugal
- Was used as reference implementation in Cisco R&D labs
- High interest from major telco players
  - Cisco, Comcast, HP, Intel, Motorola, Nokia, Xerox
- Open source
- Zero budget

Dibbler :: Current Status (2)

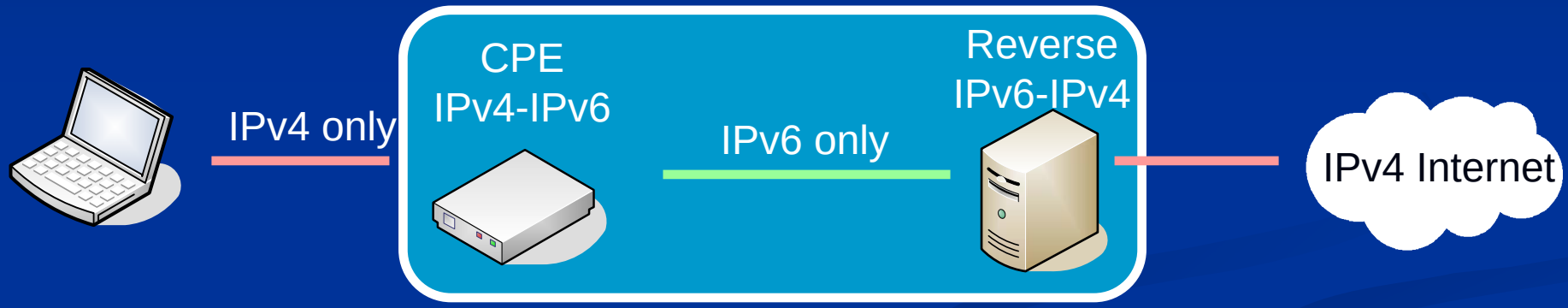
- Available for:
  - Linux (Debian, Ubuntu, Gentoo, PLD, OpenWRT, sources)
  - Windows NT, 2000, XP, 2003, Vista
  - MacOS (work in progress)
- Ported to various architectures
  - x86, x86_64, amd64, Sparc, PowerPC, Alpha (64bit RISC), Motorola m68k
  - MIPS (embedded system, LinkSys wireless router WRT54)
  - IBM S390 (mainframe class)

Dibbler :: Some features

- Stateless DHCPv6
  - Just some options, not an adress or prefix
  - Also anonymous
- Client classification
  - e.g. VoIP phones and PCs in one network, each receiving addresses/prefixes from a different pool
  - Exceptions
    - This client should receive some non-standard options
- Parameter negotiation
  - Clients are allowed to send values as hints
- Integration (IPAM style)
  - XML files exported
  - Leasequery protocol supported (RFC5007)
  - GNU GPL licence
- Inteoperability
  - Client is able to transmit any option

Dibbler :: Non-standard extensions

- Remote IPv4-to-IPv6 NAT/tunneling
  - A box with:
    - IPv4-to-IPv6 NAT
    - IPv4-over-IPv6 tunneling
  - Remotely configurable
  - Developed Linux kernel to do the IPv4-to-IPv6 NAT



- Address with addresses assignments
  - Extra suboption for specifying prefix length for assigned address
- Authentication
  - All transmissions fully authenticated
  - keys based (SHA1, SHA256, SHA512)
  - Manual key distribution
- Mobile DHCPv6
  - How to obtain addresses and options remotely?

DHCPv6 interop events

Interop events

- Associated with IETF (before or after)
  - RIPE, Amsterdam, ~ 68th IETF meeting
  - Vancouver, IETF 70th meeting
  - Comcast, Philadelphia, IETF 71th meeting
- ~7-10 vendors participated in every event
- Details are not to be disclosed
- Major vendors:
  - Cisco, RedHat, Sun, HP, Microsoft, Nokia
- Telcos:
  - Comcast, Telus
- And several other vendors:
  - Weird Solutions (Sweden), Nominum
- Open-source:
  - ISC, Dibbler

Common issues

- Top issue #1
  - DHCPv6 assigns addresses and options
  - 2 clients after receiving address from the same server using the same pool can't ping each other
  - Address assigned only, without associated prefix
  - There is no way to transmit that information over standard DHCPv6 message
  - Prefixes to be configured via RAs
- Conclusion:
  - Flaw/design assumption in DHCPv6 protocol. It cannot without routers.
- Solution:
  - So common, that 2 workarounds have been implemented:
  - Assign address with /64 prefix (arbitrary chosen length)
  - Suboption proposed and implemented

Interop issues

- FQDN
  - Some vendors provide support, but the actual update is done over IPv4
  - Bugged implementation in latest popular operating system (=interop issues)
    - Confirmed and reported by Xerox and others, no response
- Flawed handling of non-empty IA options
  - Fixed in several implementations for better interop
- Leasequery:
  - Recent addition (RFC5007, Sept. 2007)
  - Few implementations and bugged
  - Rapid improvements
  - Extra features being standarized

Q&A

Thank you

Please visit Dibbler homepage:
http://klub.com.pl/dhcpv6/

Tomasz Mrugalski
Gdansk University of Technology, Poland