Communication Networks 2
Exercise 6 - E-Mail

TECHNISCHE UNIVERSITÄT DARMSTADT

Multimedia Communications Lab
TU Darmstadt

Problem 1 MIME

What does the abbreviation “MIME” stand for?
◯(A) Multiple Internet Mail Extensions
◯(B) Multipurpose Internet Mail Extensions
◯(C) Mail in Mail Encapsulation
◯(D) Mail Internet Message Extension
◯(E) Mandatory Internet Mail Encryption

Solution:
◯(A) Multiple Internet Mail Extensions
⊗(B) Multipurpose Internet Mail Extensions
◯(C) Mail in Mail Encapsulation
◯(D) Mail Internet Message Extension
◯(E) Mandatory Internet Mail Encryption

Problem 2 MIME-Cont.

Which of the following is *not* a MIME extension?
◯(A)Model
◯(B)Text
◯(C)Application
◯(D)Segment
◯(E)Message

Solution:
◯(A)Model
◯(B)Text
◯(C)Application
⊗(D)Segment
◯(E)Message
Defined types are Application, Text, Multipart, Message, Image, Audio, Video and Model; see RFC 2046 section 3 and RFC 2077

Problem 3 Mailservers

Which protocol is used for the communication between mailservers (MTAs)?
◯(A)POP3 and HTTP
◯(B)IMAP and SMTP

◯(C)HTTP
◯(D)SMTP
◯(E)SMTP and ESMTP

Solution:
◯(A)POP3 and HTTP
◯(B)IMAP and SMTP
◯(C)HTTP
◯(D)SMTP
⊗(E)SMTP and ESMTP

Problem 4 Email Adress

Which of the following is *not* a valid email address?
◯(A)user@example.net
◯(B)> ^ „ ^ <@meow.example.org
◯(C) ^ _ _ ^ @example.com
◯(D)user@äöüß.example.org
◯(D)"> ^ „ ^ <"@purrrr.example.org
◯(E)user@[127.0.0.1]

Solution:
◯(A)user@example.net
⊗(B)> ^ „ ^ <@meow.example.org
◯(C) ^ _ _ ^ @example.com
◯(D)user@äöüß.example.org
◯(D)"> ^ „ ^ <"@purrrr.example.org
◯(E)user@[127.0.0.1]
“>”, “<” and “,” are not RFC2822 atext tokens and thus need to be quoted (cf. E)

Problem 5 Email Message

You receive a message with a Subject:-header of “=?utf-8?b?4piD?=”. Which season is it?
◯(A) Spring
◯(B) Summer
◯(C)Autumn
◯(D)Winter

Solution:
◯(A) Spring
◯(B) Summer
◯(C)Autumn
◯(D)Winter
E). Base64-decoding “4piD” leads us to the bytes 0xe2, 0x98 and 0x83 which are the UTF-8 representation of the Unicode snowman character (U+2603). Online Subject Decoder

Problem 6 Email Message-Cont.

A binary file is 3072 bytes long. How long will it be if encoded using base64 encoding, with a CR+LF pair inserted after every 80 bytes sent and at the end?

Solution:

a) Base64-Encoding performs a character encoding from 8bit to 6bit

b) The encoding will break the message into 1024 units of 3 bytes each.

c) Then line breaks must be added after every 80 bytes

d) In sum:
   + 4096 bytes because each of these units be encoded as 4 bytes
   + 104 bytes for 52CRs + 52 LFs for 52 lines with 80 chars each
   = 4200 bytes.

Problem 7 SMTP

How does SMTP mark the end of a message body?
Hint:See RFC5321

Solution:

- The character sequence "<CRLF>.<CRLF>" ends the mail text.
- But this sequence cannot be sent by the user
- To allow the usage of such “forbidden” sequences following is used:
  Client: Before sending each line: If the first char it is a period, add one additional period at the beginning.
  Server:After receiving a line of text: Check if the line consists of a single period. If yes, the message is finished. If not, remove the period.

Problem 8 Email Delivery process

Perform all steps necessary to manually send a minimal email to kn2_practical_task@kom.tu-darmstadt.de from outside the TUD network. Which mailservers are involved in the delivery process? What steps do you need to take? Which command line programs can you use? Can you choose arbitrary senders?

Solution:

a) Lookup the DNS MX records for kom.tu-darmstadt.de, for example using

b) nslookup -query=MX kom.tu-darmstadt.de

c) note the different preference numbers and that only mailin.hrz.tu-darmstadt.de is reachable from the internet

d) Connect via TCP to the mailserver on port 25:

e) nc mailin.hrz.tu-darmstadt.de 25

f) Send the message, use the SMTP commands HELO, MAIL FROM, RCPT TO, DATA and QUIT:

- HELO myhostname<CRLF>
- MAIL FROM:<irina.diaconita@kom.tu-darmstadt.de><CRLF>
- RCPT TO:<kn2_practical_task@kom.tu-darmstadt.de><CRLF>
- DATA<CRLF>
- From: irina.diaconita@kom.tu-darmstadt.de<CRLF>
- To: kn2_practical_task@kom.tu-darmstadt.de<CRLF>
- Subject: Bringing cookies tomorrow!<CRLF>
- <CRLF>
- Chocolate ones! A lot!<CRLF>
- .<CRLF>
- QUIT<CRLF>

Problem 9 Email Threats

A mail is sent from your laptop via SMTP to your ISP and from there on to a friend who uses Google Mail on the web. Someone is really interested to read the contents of that email. Which mechanisms can prevent that and against which threats are they effective?

Solution:

a) Transport-layer encryption using STARTTLS
   - Effective against passive attackers on the wire, typically not authenticated and not enforced
   - Not effective against interception at rest in different places

b) Application-layer encryption using PGP or S/MIME
   - Effective against active attackers on the wire
   - Not effective against compromised machines/keys

Problem 10 Email Inline Image

You want to send a HTML mail with an inline image. What are different ways to do this? What are advantages and disadvantages?

Solution:

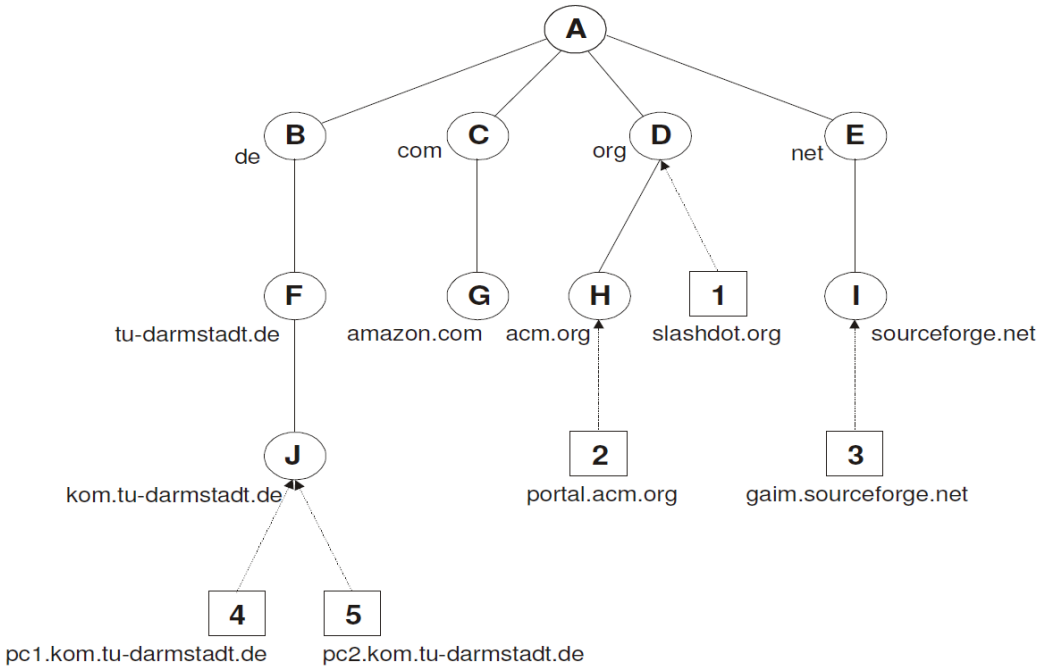
a) <img src=http://example.org/kitten.jpg>, Advantage: no additional bytes in the mail, Disadavantage: Privacy

b) <img src=cid:kitten.jpg>, Attachment with Content-ID: kitten.jpg.

c) <img src=...>, Advantage: no external network connection needed/privacy, Disadavantage: might get big, overhead for Base64-encoding

Problem 11 DNS Addressing

Assume the scenario depicted below. The scenario consists of a hierarchy of 10 name servers **(A-J)** and 5 hosts **(1-5)**, which want to resolve names. Server **A** is the root server in this scenario. In addition, server **B** is the only server in this scenario that is able to cache names. It has already cached all names of the net-domain. The other servers are not able to cache names and, therefore, normally forward non-authoritative requests (requesting resolution of adresses which are not within their own domain) directly to the root if not explicitly configured in another way. In this scenario, Server **H** forwards all non-authoritative requests to name server **D**. All name servers in the domain **tu-darmstadt.de** (and its subdomains) are configured to forward all non-authoritative requests directly to server **B**.
The requests are executed in temporal sequence. The dashed lines show which name server is used by a host for name resolution.



Please name all DNS servers used for the following name resolution requests of the 5 hosts (1-5):

Host 1 → www.amazon.com

Host 2 → gaim.sourceforge.net

Host 3 → e-technik.tu-darmstadt.de

Host 4 → info.net

Host 5 → informatik.tu-darmstadt.de

Solution:

Host 1: D, A, C, G, www.amazon.com

Host 2: H, D, A, E, I, gaim.sourceforge.net

Host 3: I, A, B, F, e-technik.tu-darmstadt.de

Host 4: J, B, info.net

Host 5: J, B, F, informatik.tu-darmstadt.de

Problem 12 DNS Hierarchy

What are the four different typical levels of a DNS hierarchy? Explain the tasks at each level.

Solution:

a) DNS root name servers: There are 13 root servers worldwide. They are contacted from local name servers, that cannot resolve a name. The root server contacts an authorative name server, if the mapping is not already known. At last the local name server gets the mapping.

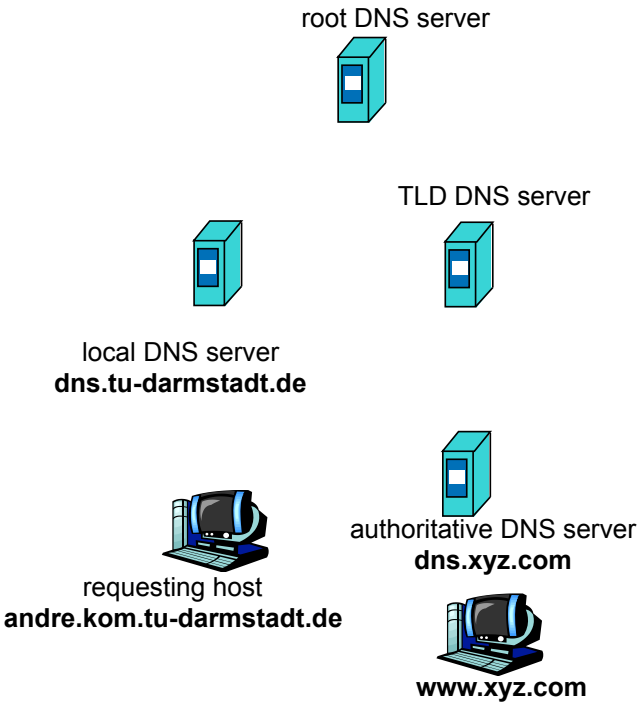
b) Top-level domain (TLD) servers: They are responsible for the top level domains like .com, .org, .net, and the top-level country domains like .de, .jp.

c) Authoritative DNS servers: They are the DNS server of organizations and provide mappings for the organization's servers like mail. They are maintained either by the organization themself or by a service provider.

d) Local name server: They do not strictly belong to the hierarchy. Each ISP has at least one, the default name server. They act as proxy for DNS queries and forward them into the hierarchy.

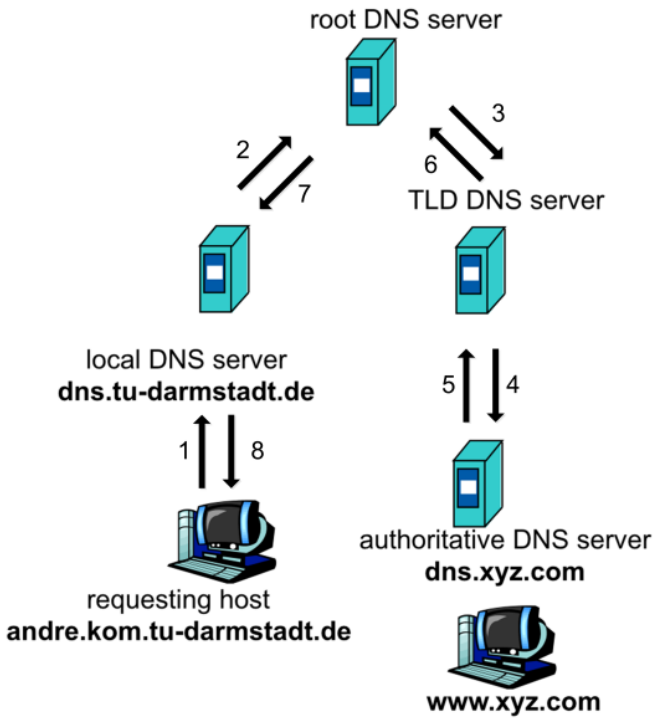
Problem 13 DNS Name Resolution Process

Show the name resolution process of the host andre.kom.tu-darmstadt.de asking for www.xyz.com in the figure below. Assume that name resolution is performed fully recursive.



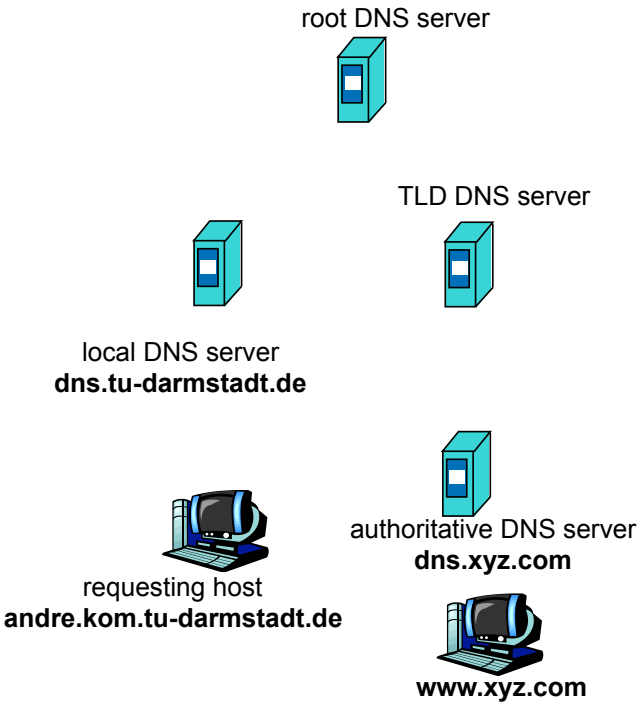
Solution:

Remember: Recursive means the query will be processed and queried further by each DNS server.



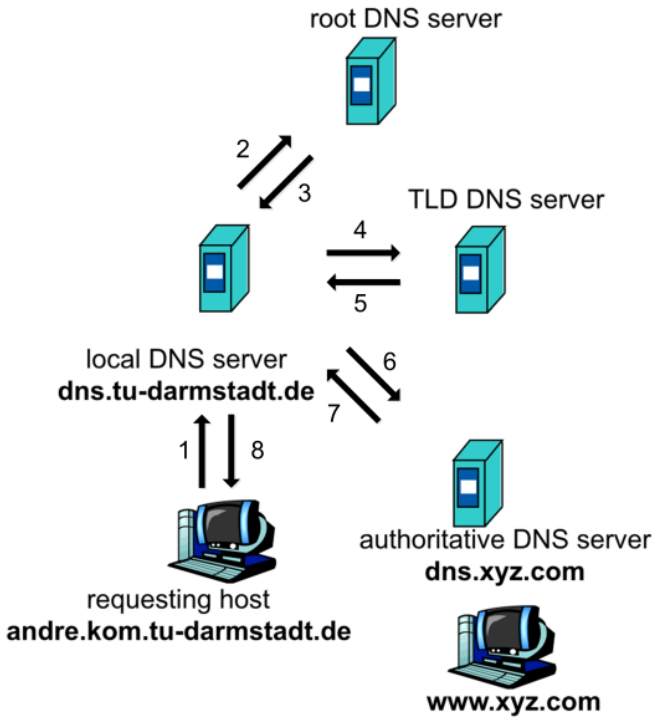
Problem 14 DNS Name Resolution Process- Cont.

Show the name resolution process of the host andre.kom.tu-darmstadt.de asking for www.xyz.com in the figure below. Assume that name resolution is performed iteratively starting from the local name server.



Solution:

Remember: Iterative means the query will be answered only partially, if no mapping is available.

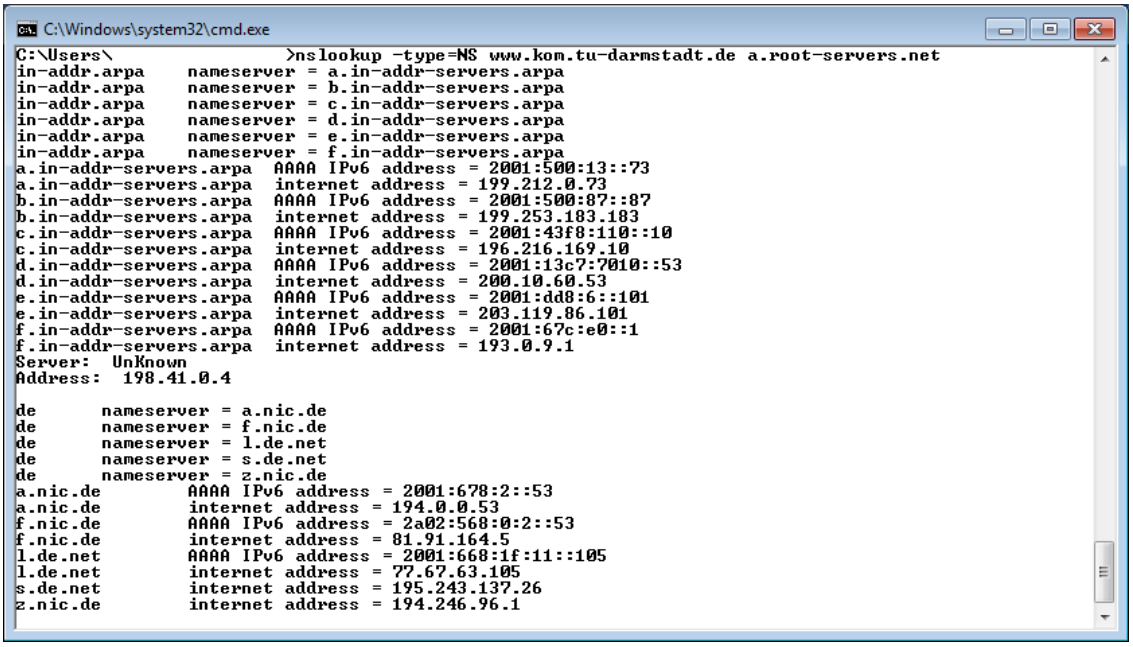


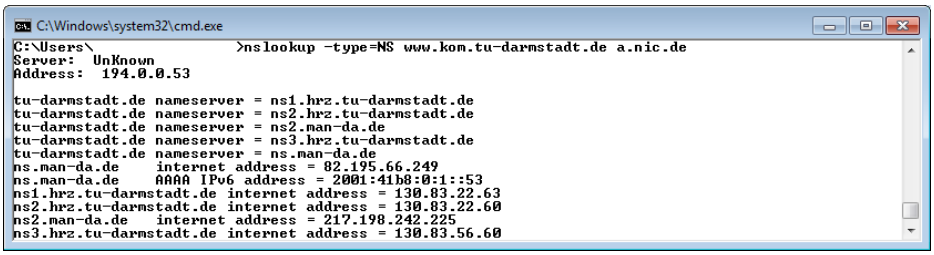
Problem 15 DNS Nslookup

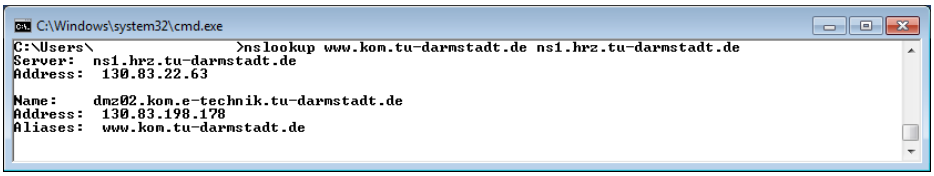
Use nslookup to perform the address resolution of a Web site of your choice. Start from one of the root name servers [a-m].root-servers.net and perform an iterative query as your local name server would do.

Solution:

nslookup is a tool for getting the IP address of servers using DNS. For the different available nslookup types see `http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/nslookup_set_type.mspx`







Problem 16 DNS Nslookup-Cont1.

Use nslookup on your local host to send DNS queries to three DNS servers: your local DNS server and the two DNS servers you found in the previous task. Try querying for Type A, NS, and MX records. Summarize your findings.

Solution:

nslookup -type=A www.tu-darmstadt.de
nslookup -type=NS www.tu-darmstadt.de
nslookup -type=MX www.tu-darmstadt.de
...

Problem 17 DNS Nslookup- Cont.2

Use nslookup to find a Web server that has multiple IP addresses. Does the Web server of TU Darmstadt have multiple IP addresses?

Solution:
nslookup www.google.com → multiple IP addresses
nslookup tu-darmstadt.de → single IP address

Problem 18 DNS Whois Database

Use various whois databases on the Internet to obtain the names of two DNS servers. Indicate which whois databases you used.

Solution:
E.g. `http://www.nic.com/nic/whois/` and `http://www.denic.de/de/hintergrund/whois-service.html` using an arbitrarily domain, www.google.com or www.tu-darmstadt.de

Problem 19 DNS Whois Database- Cont.1

Use the appropriate whois database to determine the IP address range of TU Darmstadt.

Solution:
`https://apps.db.ripe.net/search/query.html` with 130.83.58.211 → 130.83.0.0 – 130.83.255.255