

DNS and SMTP

1 DNS Commands

`dig` and `host` are two commands which allow you to make DNS requests on the command line. Especially, `dig` returns a list of root servers when called without parameter. Read their man page to learn about their syntax, in particular how to specify which DNS server to contact, how to query NS records instead of the default A record, and how to make non-recursive query.

Once you know how to do it, perform recursive queries by hand on the following domain names: `www.mit.jyu.fi`, `microsoft.com`, `www.ic.ac.uk`.

Then perform, again by hand, a recursive reverse query on the following IP addresses: `209.85.137.19`, `88.191.250.104`, `192.58.128.30`.

2 E-mail

2.1 With the mail Command

In the Good Old DaysTM, mail was sent and read using the `mail` command. You will not be able to read your e-mail with the `mail` command, since it expects to find the mail in a local mailbox, but sending is possible. Read the man page and send an e-mail to yourself while looking at the exchange between `mail` and the server using Wireshark. Once the e-mail is sent, access it through your usual tool and take a look at the headers. What path did it follow until it reached your mailbox ?

2.2 With nc

Now use `nc` to connect to the university's SMTP server (`smtp.jyu.fi`, port 25) and send by hand an e-mail to yourself following the SMTP protocol. Do this twice, first by using the old RFC 821 `HELO` command, and a second time using the extended RFC 2821 `EHL0` command. Do you see any difference?

Now you don't actually need to give your real address in the `MAIL` command. You can put whatever address you like. Try to send yourself an anonymous e-mail. Can you use the university's SMTP server to send e-mail to an address outside of the university's domain (if you own such an address)? What if you specify a domain that does not exist at all (e.g., made of completely random letters) as the sender's domain?

When you send e-mail using a fake sender's address to yourself, how much information about the real sender can you gather from the headers in the e-mail message?

3 Home Exercise

In order to pass the course, you need to develop, as a final assignment, a software which includes a protocole implementation; you will also demonstrate your application to the teacher. Note that if you do not submit a topic proposal, your assignment will not be validated.

Depending on your skills and motivation, you may choose between three types of software:

1. Implement a client for a simple, well known protocol, such as POP3, DICT, HTTP... If you choose this path, the grading of your assignment will be very strict, as it is a simple task.

2. Write a gateway software which consists of one server running one standard protocol, and a server or a client running another standard protocol, your software being a bridge between them. The software doesn't necessarily need to make much sense, as long as the protocols are implemented correctly, and that the data flows as expected from one side to the other. You can for example make an IRC to HTTP gateway, or IRC to SMTP... On this path, the grading will be a bit less strict, since it is a more difficult task.
3. Write a client and a server using a protocol of your own design, and write as well the specifications for that protocol. The protocol doesn't need to run directly over UDP or TCP, it may also use another existing protocol as its transport; in this case you are allowed to reuse existing software modules for implementing the underlying protocol. On this path, the grading will also be less strict than in the first one.

Your task for next week is to write a short description of the kind of assignement you want to make, and send it to the teacher for acceptance. Please send your description to `mweber@mit.jyu.fi` before 12.2.2009. Acceptable formats are plain text and PDF. Don't hesitate to ask questions about the assignment if something is unclear.