Computer Networks
Jacobs University Bremen
Dr. Jürgen Schönwälder

Course: 320301
Date: 2015-11-27
Due: 2015-12-06

Problem Sheet #6

Problem 6.1: *IP over DNS tunnel* (10 points)

In this assignment, you will extend the IP over TCP tunnel (see previous assignments) to send IP packets over DNS. The idea is to turn an IP packet received from a Tun interface into a set of DNS queries that are sent to the remote tunnel endpoint. The remote tunnel endpoint sends a DNS response to indicate the receipt of the DNS query.

Since IP packets are larger than the amount of data carried in a typical DNS query (note that DNS labels are limited to 63 octets and query names to 255 octets), it is necessary to fragment a received IP packet and to send the fragments in a sequence of DNS queries. You may want to use some of the DNS transaction ID bits to carry a packet identification number and you may use other fields of the DNS packet (e.g., the query type field) to indicate the fragment number or fragment offset.

The receiver needs to maintain IP packet reassembly buffers. Once a packet has been reassembled successfully, the packet is delivered back into the networking stack. The receiver needs to maintain timers in order to reclaim packet reassembly buffers. It also may indicate in DNS response messages which fragments are missing so that the other tunnel endpoint can resend fragments.

Some additional hints:

- In order to generate DNS labels that consist of case-insensitive 7-bit ASCII characters, you may consider using base-32 encoding as defined in RFC 4648 [1]. If you do not mind being slow, you could also use a dictionary of words to encode bits (e.g., each octet is mapped to a word in a dictionary of popular 3-letter acronyms).
- Avoid generating bursts of DNS queries. Instead, pace the queries by waiting a randomized amount of a few milliseconds between each DNS query.
- You are welcome to use DNS library functions to encode or decode packets. The `bind`[1] implementation is widely available but a bit client-centric. A never DNS implementation that comes with proper libraries is called `unbound`[2].

Make sure that tools like `wireshark`[3] can properly decode your DNS packets.

References

[1] S. Josefsson. The Base16, Base32, and Base64 Data Encodings. RFC 4648, SJD, October 2006.

[1] https://www.isc.org/downloads/bind/
[2] http://unbound.nlnetlabs.nl
[3] https://www.wireshark.org/