Stress-tests procedures for a DNS node (BIND and NSD)
The main purposes of the stress-tests were:

- to find out a maximum performance that is provided by DNS software in case of given hardware configuration

- time measurements of AXFR- and IXFR-transfers in case of given DNS software for the given zone files
Sandbox overview

- ethernet switch that supports 1000baseT, 10Gbase-SR and 802.3ad modes
- two servers with Intel Xeon CPU, 6 cores with HT-feature, two onboard gigabit Ethernet ports Intel i350 on both servers and one of the servers has 10 gigabit Ethernet card Intel Converged Ethernet adapter X520-LR1.
- FreeBSD 10.3-RELEASE operating system is installed on both servers, the software dnsperf-2.0.0.0 and tcpreplay-4.1.0 (with /dev/netmap support) was used for generating DNS-queries. A special Python-script was written to measure the number of DNS-replies received by the service.
Sandbox network topology

Node generating stress-tests
- CPU: Xeon, 2.10GHz, 6 cores with HT
- RAM: 32768 MB
- 2x1-gigabit/lmax 10GE network interfaces
- OS: FreeBSD 10.3-RELEASE/Custom kernel (netmap support)

Node under stress-tests
- CPU: Xeon, 2.40GHz, 6 cores with HT
- RAM: 16GB
- 2x1-gigabit/lmax 10GE network interfaces
- 1x10-gigabit/lmax QS20 network interface
- OS: FreeBSD 10.3-RELEASE/GENERIC kernel

1000base-T link

10Gbase-SR link
Basic actions

- For the first test we configured DNS-zone of IP6.ARPA. with 3 millions PTR records (DNSSEC was not implemented)
- For the second test we configured a copy of RU. zone with 5 millions records of A-, AAAA-, NS-types and also the RRsets from DNSSEC specification
- We collected the RU-zone queries into single pcap-file (hour interval)
- We executed stress-tests (IP6.ARPA and RU-zone) for BIND 9.9.9-ESV software
- We executed stress-tests (IP6.ARPA and RU-zone) for NSD 4.1.7 software
stress-test - service overload

BIND: a series of dnsperf-processes was launched on the server. Summary queries rate was: 400 kqps, answers rate was: 150 krps. Channel utilization:

NSD: a single tcpreplay-process through /dev/netmap interface was launched on the server. Queries rate was: 1000 rqps, answers rate was: 550 krps. Channel utilization:
stress-test - maximum service load with no quality loss

BIND: a series of tcpreplay-processes was launched on the server, queries bandwidth was fixed for each process, summary queries rate was at a boundary value when qps was equal rps with no packet loss. Queries rate was: 120 kqps, answers rate was: 120 krps. Channel utilization:

NSD: a series of tcpreplay-processes was launched on the server, queries bandwidth was fixed for each process, summary queries rate was at a boundary value when qps was equal rps with no packet loss. Queries rate was: 510 kqps, answers rate was: 510 krps. Channel utilization:
### Stress-tests results:

<table>
<thead>
<tr>
<th>Stress-test description</th>
<th>BIND 9.9.9-ESV (1000base-T)</th>
<th>NSD 4.1.7 (802.3ad – LACP – 2x1000base-T)</th>
<th>NSD 4.1.7 (10GBase-SR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy of zone RU. (dnssec-enabled)</td>
<td>Test zone IP6.ARPA. (dnssec-disabled)</td>
<td>Test zone IP6.ARPA. (dnssec-disabled)</td>
<td>Copy of zone RU. (dnssec-enabled)</td>
</tr>
<tr>
<td>Copy of zone RU. (dnssec-enabled)</td>
<td>Test zone IP6.ARPA. (dnssec-disabled)</td>
<td>Test zone IP6.ARPA. (dnssec-disabled)</td>
<td>Copy of zone RU. (dnssec-enabled)</td>
</tr>
<tr>
<td>Copy of zone RU. (dnssec-enabled)</td>
<td>Test zone IP6.ARPA. (dnssec-disabled)</td>
<td>Test zone IP6.ARPA. (dnssec-disabled)</td>
<td>Copy of zone RU. (dnssec-enabled)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KQPS</th>
<th>~120</th>
<th>~400</th>
<th>~540</th>
<th>~880</th>
<th>~950</th>
<th>~510</th>
<th>~1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>KPRS</td>
<td>~120</td>
<td>~150</td>
<td>~540</td>
<td>~565</td>
<td>~480</td>
<td>~510</td>
<td>~550</td>
</tr>
</tbody>
</table>

**Stress-test description**

- **BIND at a lack of CPU resources.** Series of tcpreplay-processes with fixed queries bandwidth for each process (no quality loss).
- **BIND is overloaded.** Series of dnsperf-processes flooding BIND (quality loss).
- **NSD at a lack of CPU resources.** Series of tcpreplay-processes with fixed queries bandwidth for each process (no quality loss).
- **Single tcpreplay-process with /dev/netmap interface.** Global lack of CPU on.
- **Insufficient channel bandwidth due to interface.** Series of tcpreplay-processes (quality loss).
- **NSD at a lack of CPU resources.** Series of tcpreplay-processes with fixed queries bandwidth for each process (no quality loss).
- **Single tcpreplay-process with /dev/netmap interface.** Global lack of CPU on.
Conclusions (Task 1)

Recommended use of observed DNS software on the tested server under FreeBSD RELEASE-10.3:

- BIND 9.9.9-ESV service can handle with no quality loss – 120 KQPS
  - the network is 1000base-T
    - reserved bandwidth in case of live dnssec-enabled zone is 570mbps
- NSD 4.1.7 – 510 KQPS
  - the network is 10Gbase-SR
    - reserved bandwidth - is 2,3 gbps
AXFR testing (full zone file transfer)

Initial points
- we had a set of 500000 files of tiny zones
- 200 000 zones of the set were signed
- master DNS authoritative server was BIND
- the serial-query-rate option was used on the master for AXFR rate changing
- we tested both – BIND and NSD - as a slave DNS server
- We tried to define the smallest time of AXFR transfer for the zones set
Dynamics of time change for the zones transfer via AXFR

slave BIND: minimal transfer time was about 15 minutes:

slave NSD: minimal transfer time was about 11 minutes:
IXFR testing (incremental zone files changes transfer)

We had 250 000 initial files as a set of tiny zones. Then some records were changed.

- Slave BIND: transfer time of the changes for the whole set of zones was about 18 minutes

- Slave NSD: transfer time of the changes for the whole set of zones was about 18 minutes
Conclusions (second task)

- For AXFR: minimal transfer time was achieved for the transfer of 1000 zones
- IXFR transfer time was equal for any DNS software