The impact of DNSSEC on k.root-servers.net and ns-pri.ripe.net

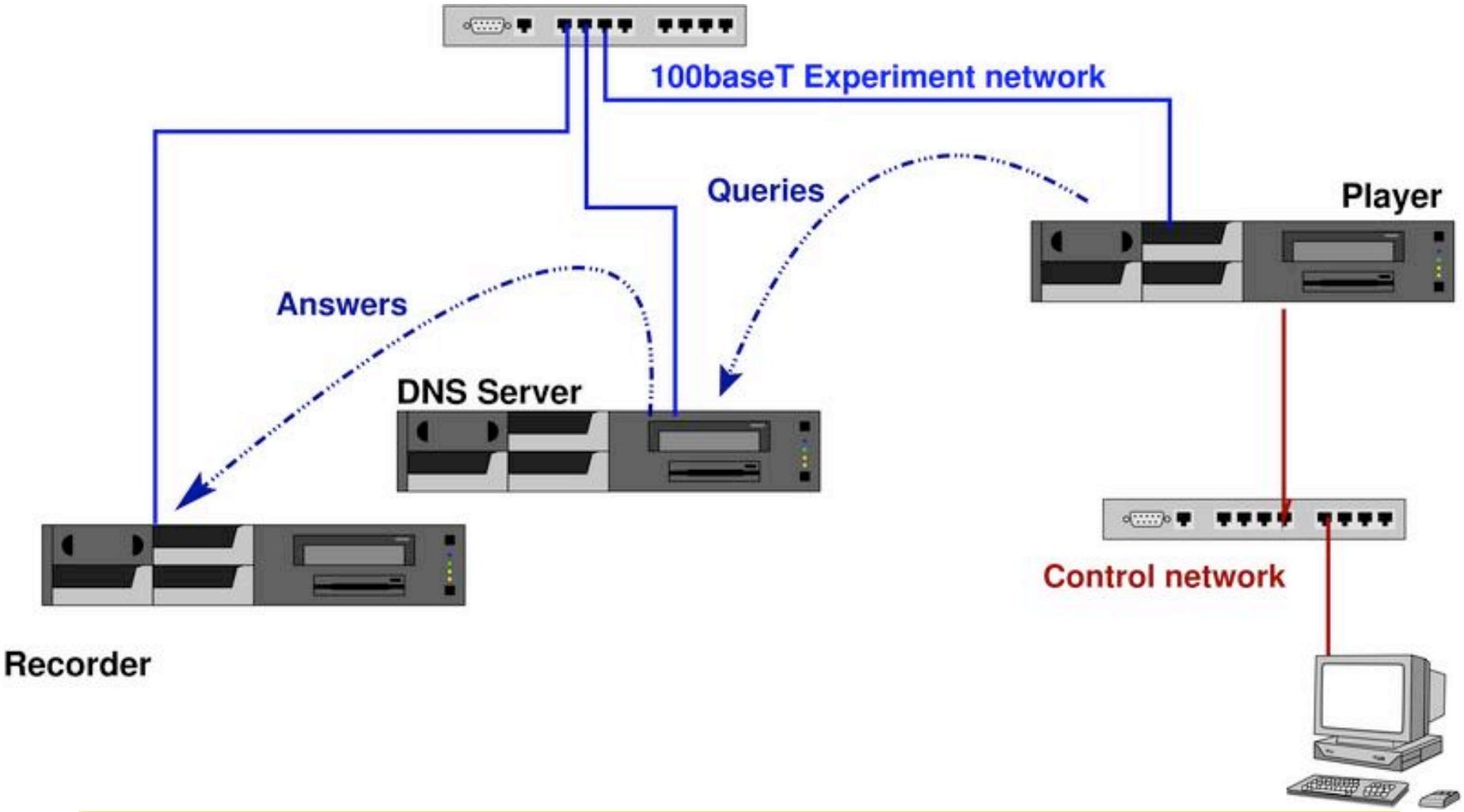
Olaf M. Kolkman

Question

What would be the immediate and initial effect on memory, CPU and bandwidth resources if we were to deploy DNSSEC on RIPE NCC's 'primary' name server?

- Measure through simulation.

The “DISTEL” Test Lab



DISTEL LAB

- Player plays libpcap traces in real time
  - libpcap traces are modified to have the servers destination address
- Server has a default route to the recorder
- Recorder captures answers

- 2 Ghz Athlon based hardware with 1 Gb memory and 100baseT Ethernet

This Experiment

- Traces from production servers:
  - k.root-servers.net
  - ns-pri.ripe.net
- Server configured to simulate the production machines.
  - ns-pri.ripe.net
    - Loaded with all 133 zones.
  - k.root-servers.net
    - Only loaded with the root zone.

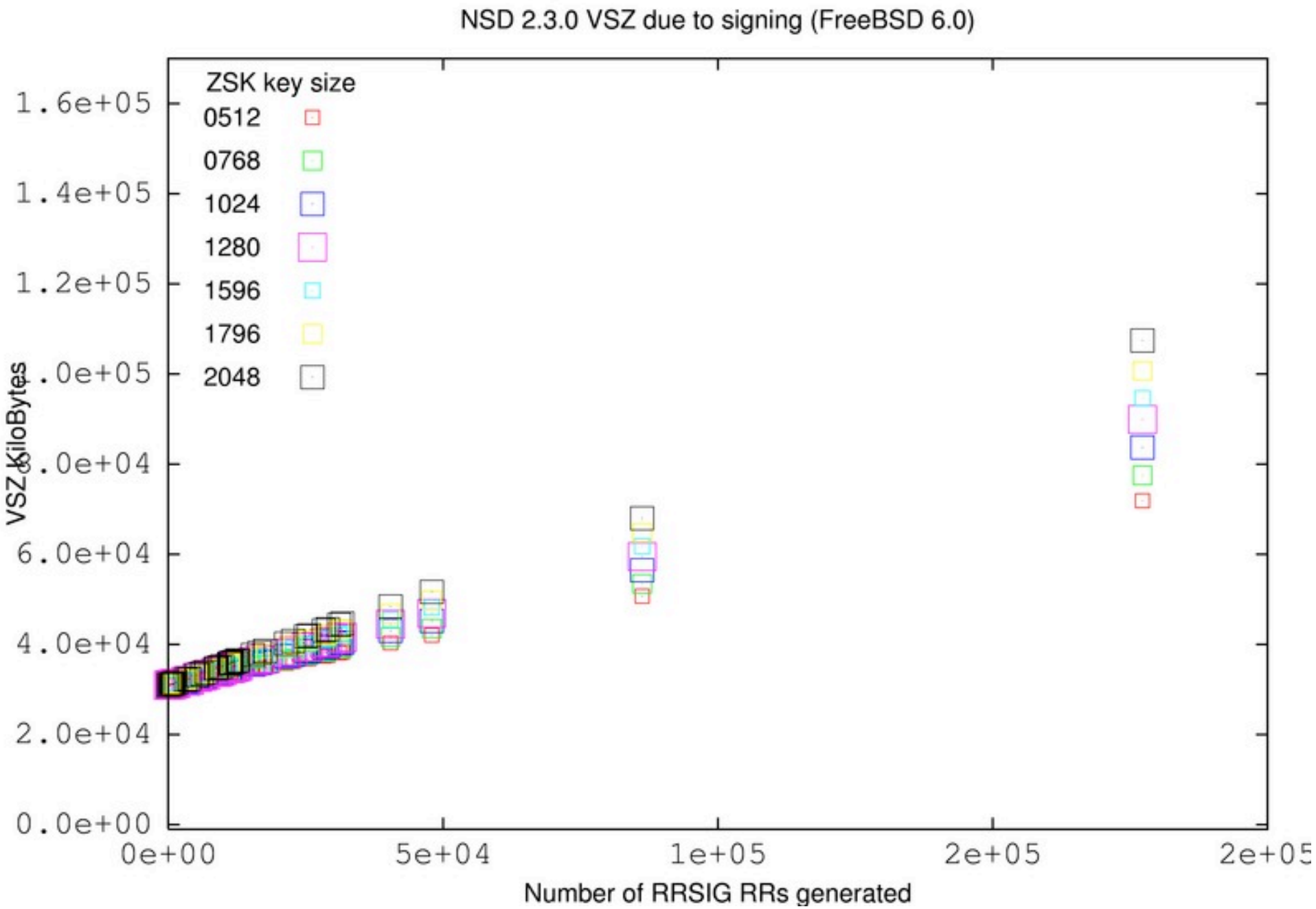
Zone Signing

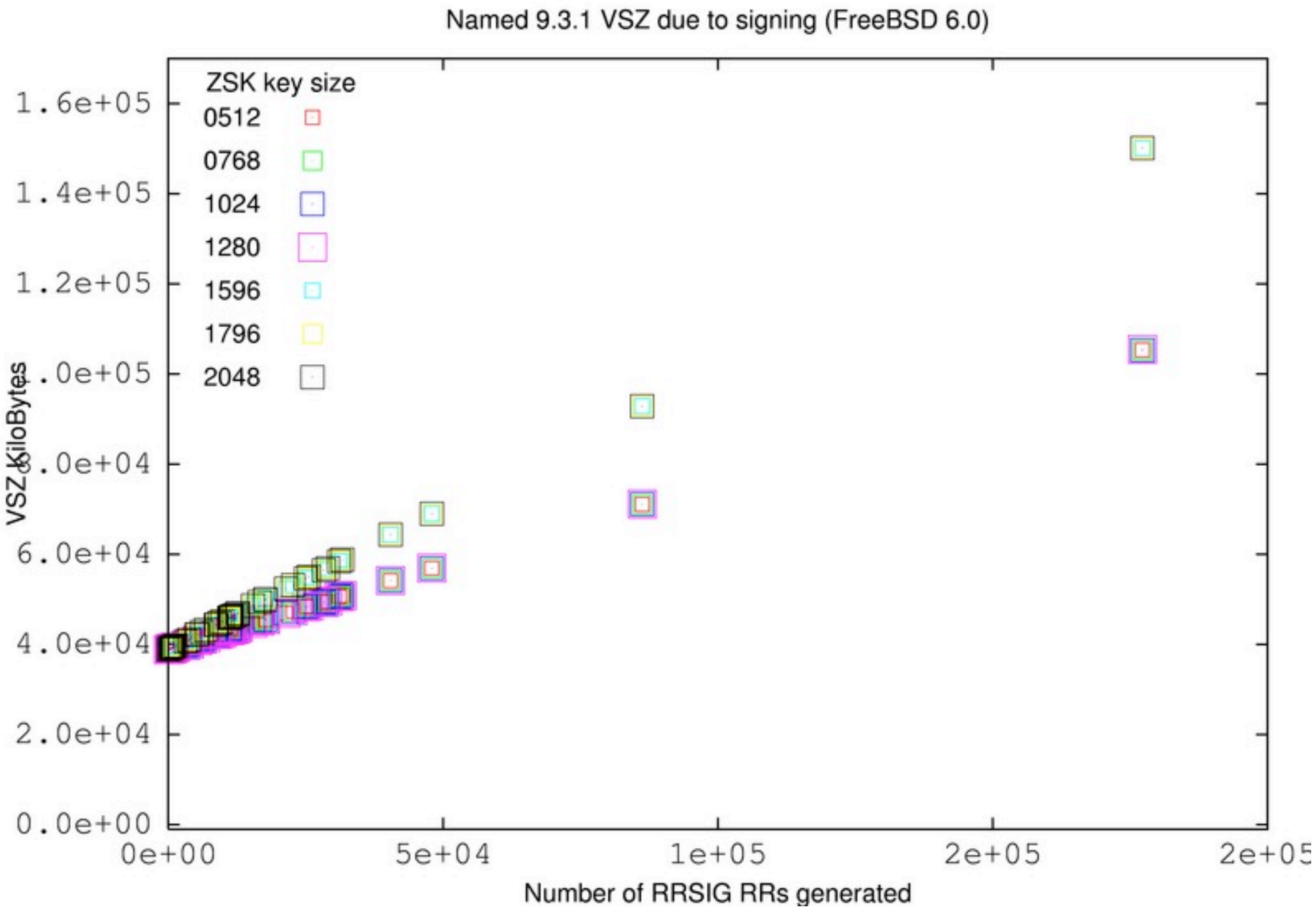
- 1 Key Signing Key 2048 bit RSASHA1
- 2 Zone Signing Keys of equal length
  - length varied between 512 and 2048
  - Only one ZSK used for signing
    - This is expected to be a common situation (Pre-publish KSK rollover)

- 3 DNSKEY RRs in per zone
  - 1 RRSIG per RR set
  - 2 RRSIGs over the DNSKEY RR set

Loading the Zones: Memory Use

- Various zone configurations were loaded.
  - Mixtures of signed and unsigned zones
  - Memory load for different numbers of RRSIGs and NSECs.
- Memory load is implementation and OS specific


NSD 2.3.0 VSZ due to signing (FreeBSD 6.0)
ZSK key size
0512
0768
1024
1280
1596
1796
2048
VSZ in KiloBytes
1.6e+05
1.4e+05
1.2e+05
1.0e+05
8.0e+04
6.0e+04
4.0e+04
2.0e+04
0.0e+00
0e+00
5e+04
1e+05
2e+05
2e+05
Number of RRSIG RRs generated


Named 9.3.1 VSZ due to signing (FreeBSD 6.0)
ZSK key size
0512
0768
1024
1280
1596
1796
2048
VSZ in KiloBytes
1.6e+05
1.4e+05
1.2e+05
1.0e+05
8.0e+04
6.0e+04
4.0e+04
2.0e+04
0.0e+00
0e+00
5e+04
1e+05
2e+05
2e+05
Number of RRSIG RRs generated

Memory

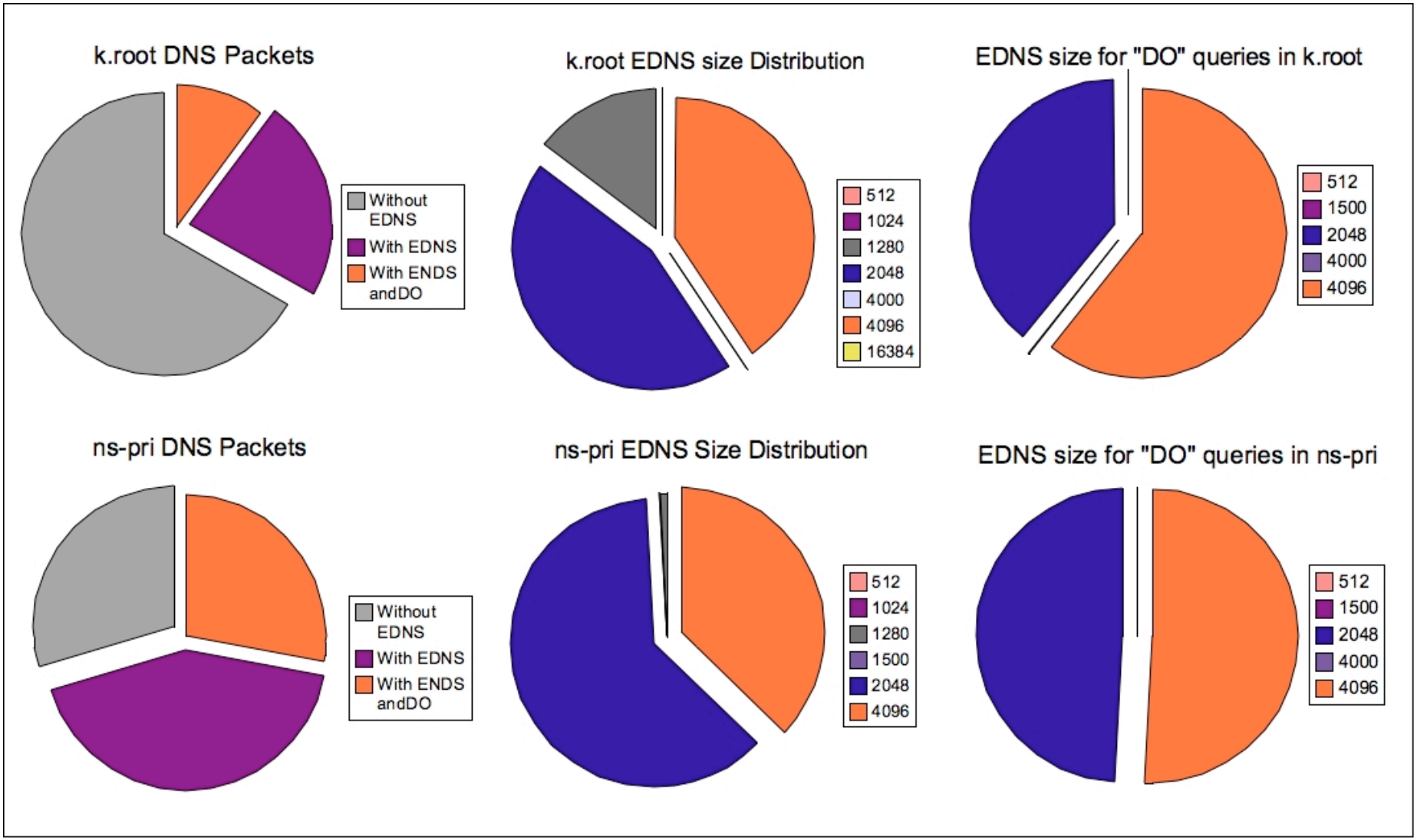
- On ns-pri.ripe.net factor 4 increase.
  - From ca. 30MB to 120MB (NSD)
  - No problem for a 1GB of memory machine
- On k.root-servers.net
  - Increase by ca 150KB
  - Total footprint 4.4 MB
- Nothing to worry about
- Memory consumption on authoritative servers can be calculated in advance.
  - No surprises necessary

Serving the zones
Query Properties

- DNS clients set the “DO” flag and request for DNSSEC data.
  - Not to do their own validation but to cache the DNSSEC data for.
- EDNS size determines maximum packet size.

  (DNSSEC requires EDNS)
- EDNS/DO properties determine which fraction of the replies contain DNSSEC information

EDNS properties



Serving the zones

- Measured for different keysizes.
  - named for ns-pri.ripe.net
  - nsd and named for ns-pri.ripe.net and k.root-servers.net

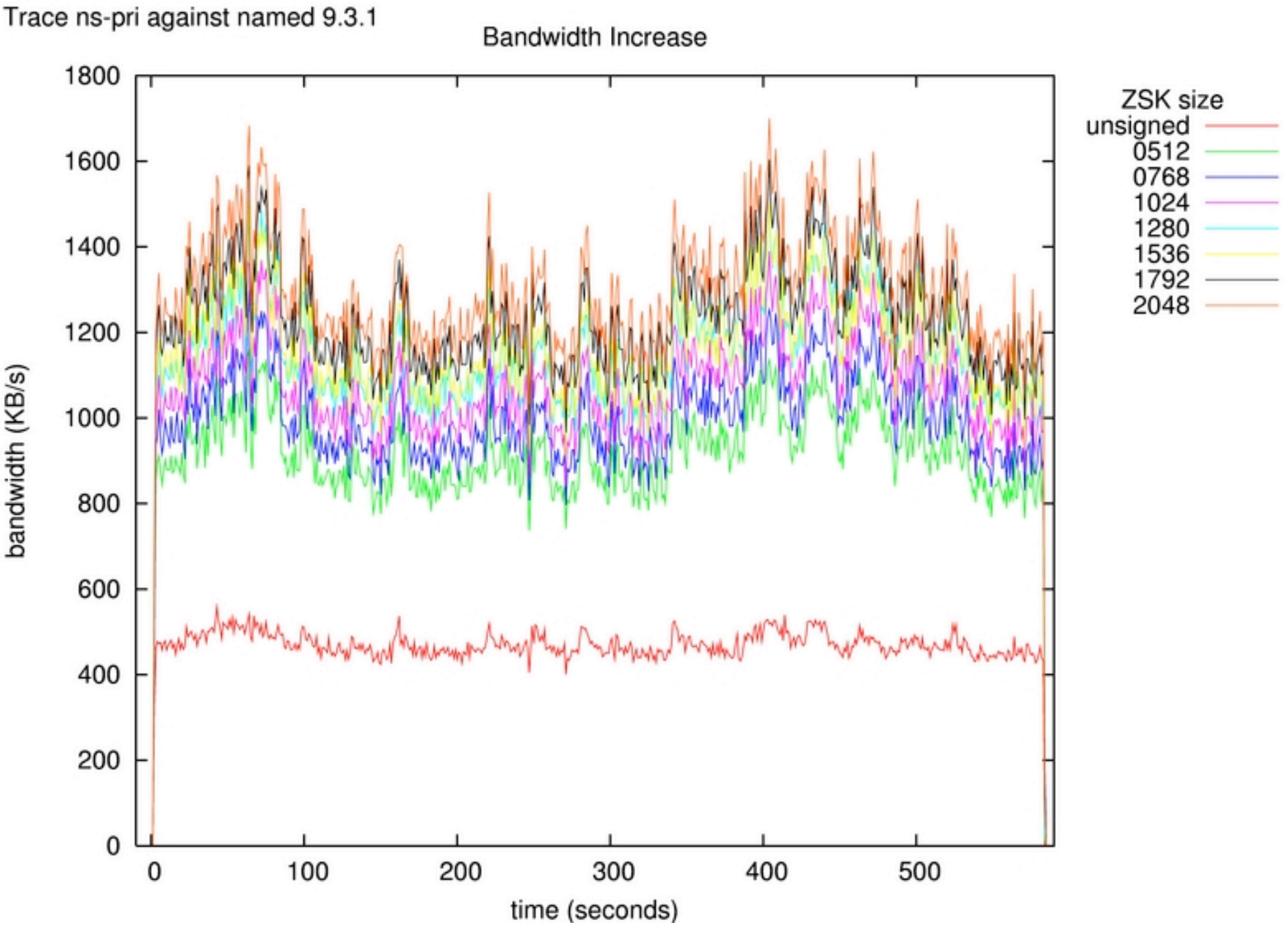
- We also wanted to study “worst case”;
  What if all queries would have the DO bit set?
  - Modified the servers to think that queries had EDNS 2048 octets size and DO bit set

CPU

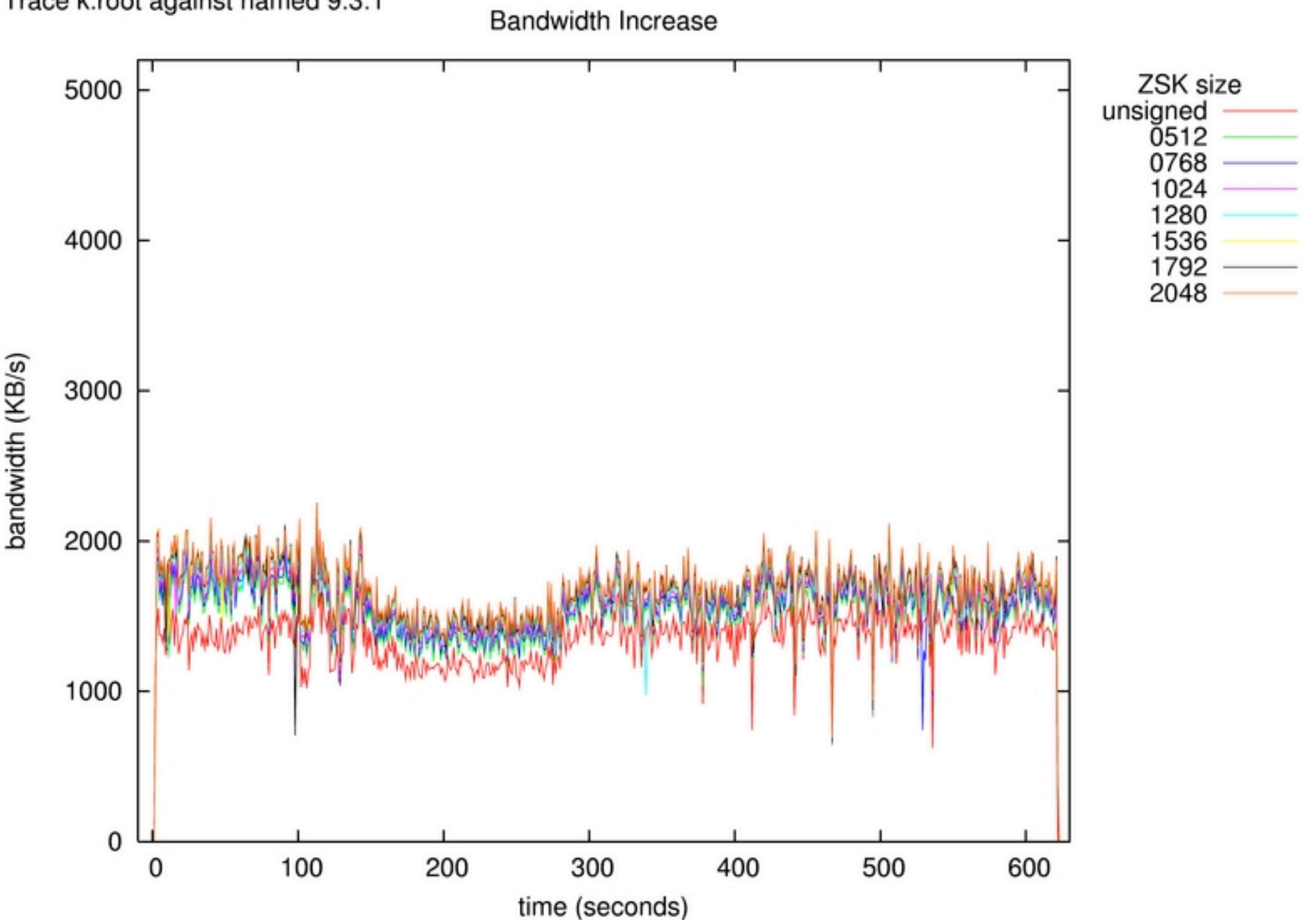
trace	server		ZSK size	WCPU
ns-pri	BIND 9.3.1		0000	ca 14%
ns-pri	BIND 9.3.1		2048	ca 18%
k.root	BIND 9.3.1		0000	ca 38%
k.root	BIND 9.3.1		2048	ca 42%
k.root	BIND 9.3.1	mod	2048	ca 50%
k.root	NSD 2.3.0		0000	ca 4%
k.root	NSD 2.3.0		2048	ca 4%
k.root	NSD 2.3.0	mod	2048	ca 5%

Bandwidth Factors

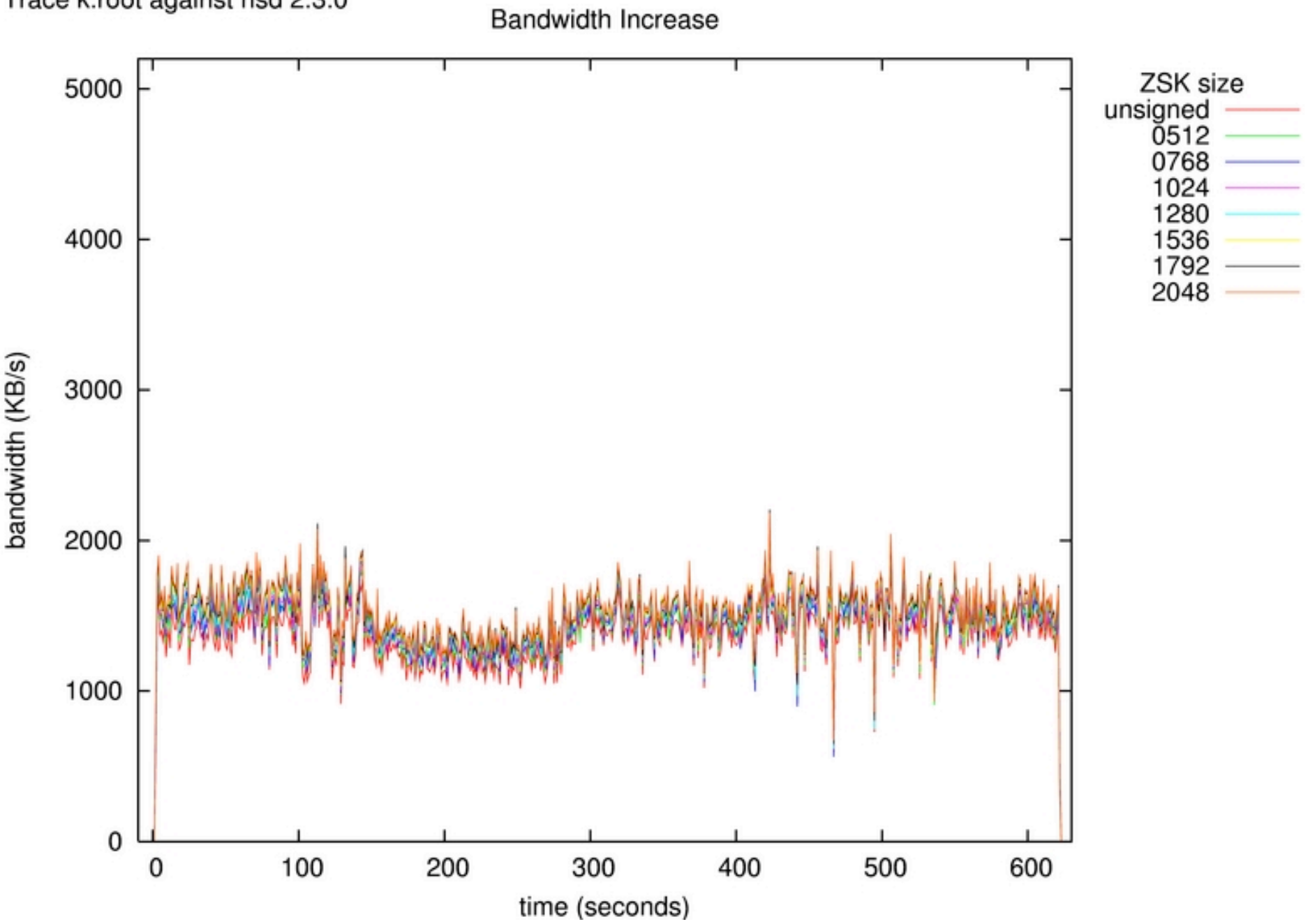
- fraction of queries with DO bit
  - Seen in difference between ns-pri and k.root result
  - Seen in difference between modified and unmodified servers
- Including DNSKEY RR in additional section.
  - Seen in difference between k.root traces from modified nsd and modified named
- Difference in answer patterns
  - Name Errors vs Positive answers
  - Difficult to asses from this data

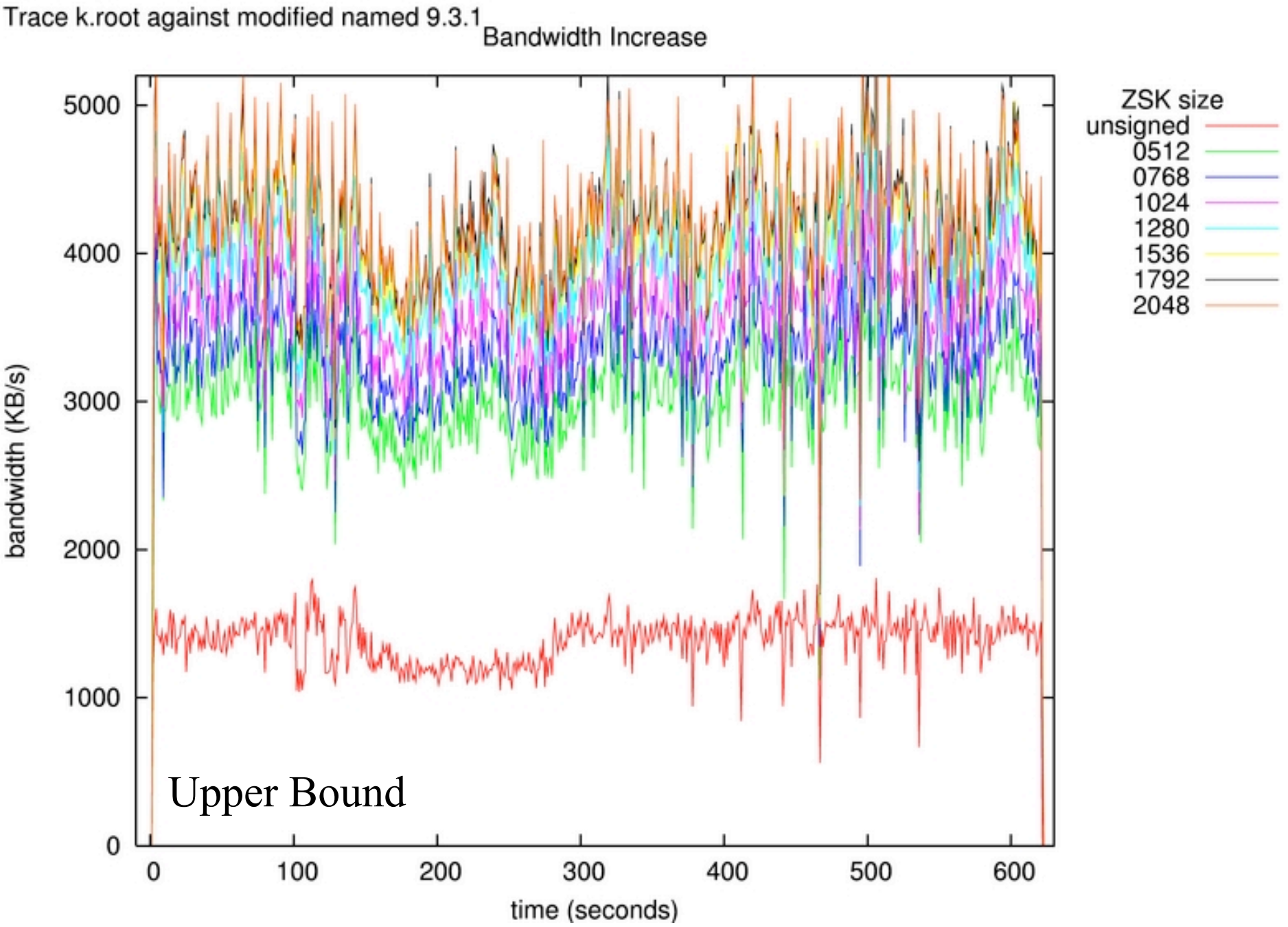

Trace ns-pri against named 9.3.1
Bandwidth Increase
bandwidth (KB/s)
time (seconds)
0
200
400
600
800
1000
1200
1400
1600
1800
0
100
200
300
400
500
ZSK size
unsigned
0512
0768
1024
1280
1536
1792
2048

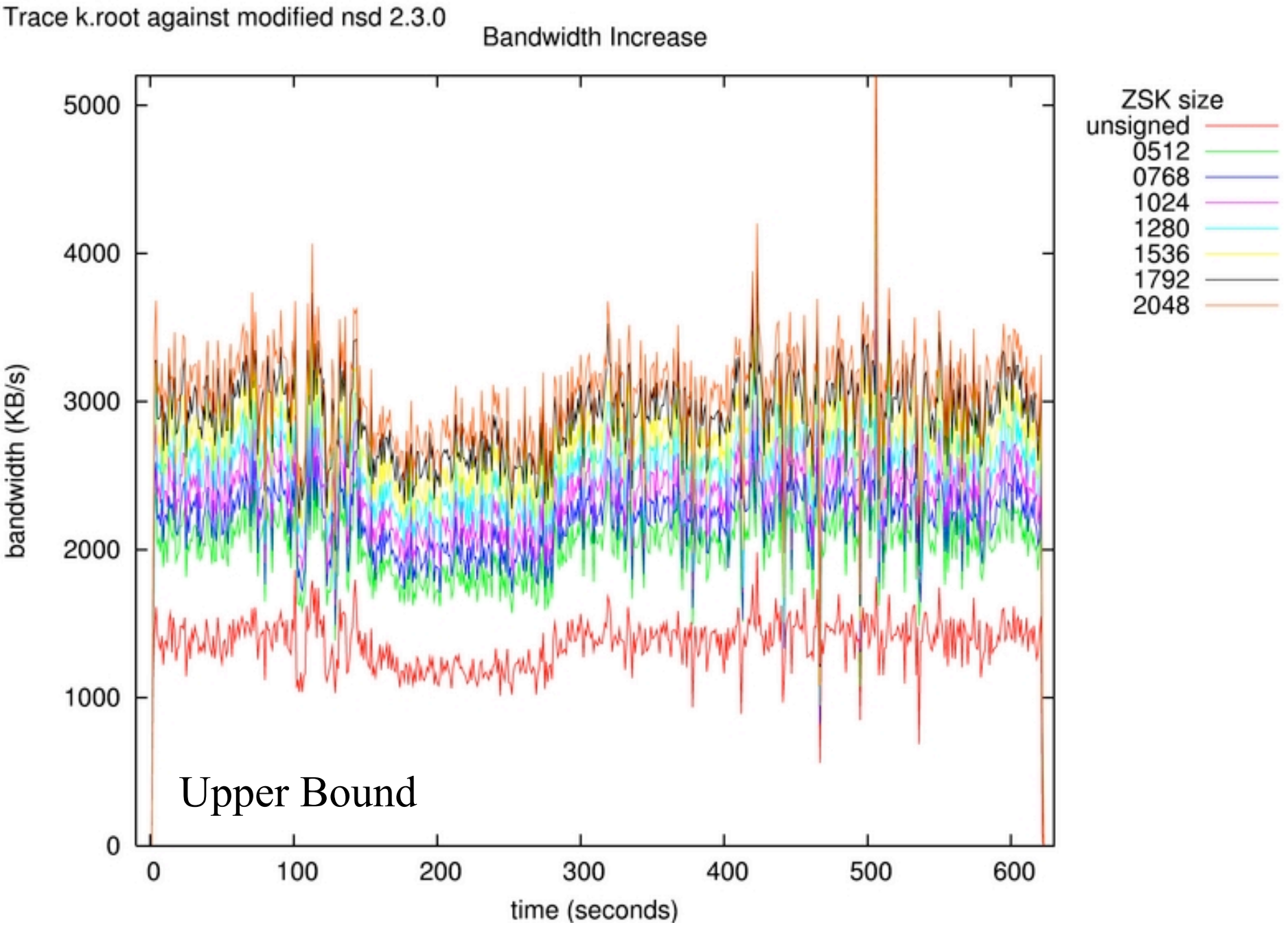
Trace k.root against named 9.3.1



Trace k.root against nsd 2.3.0




Trace k.root against modified named 9.3.1
Bandwidth Increase
bandwidth (KB/s)
time (seconds)
0
1000
2000
3000
4000
5000
0
100
200
300
400
500
600
ZSK size
unsigned
0512
0768
1024
1280
1536
1792
2048
Upper Bound


Trace k.root against modified nsd 2.3.0
Bandwidth Increase
bandwidth (KB/s)
time (seconds)
0
1000
2000
3000
4000
5000
0
100
200
300
400
500
600
ZSK size
unsigned
0512
0768
1024
1280
1536
1792
2048
Upper Bound

Bandwidth observation

- DNSKEY RR set with RRSIG in the additional section
  - Fairly big chunk of data
  - Variable size during the rollover
  - None of the clients today validate the data
  - Clients that need the data will query for it
- Servers MAY include the DNSKEY RR set
- NSD does not include
- Named does include
  - Recommendation to make the inclusion configurable

DNSKEY RR not in the additional section

- Only RRSIGs made with the ZSKs end up in the answers.
  - Usually only one RRSIG made with ZSK per RR set
  - Pre-publish key rollover grows the DNSKEY RR set but limits the amount of RRSIGs over zone content
- Validating clients will have to query for the keys
  - Break even point, where and when?

Bandwidth Increase

- Significant for ns-pri.ripe.net
  - Well within provisioned specs.
- Insignificant for for k.root-servers.net
  - Upper bound well within provisioning specs
    - even when including DNSKEY RR set in additional section

(Key size influences bandwidth but bandwidth should not influence your key size)

Conclusion

- CPU, Memory and Bandwidth usage increase are not prohibitive for deployment of DNSSEC on k.root-servers.net and ns-pri.ripe.net

- Bandwidth increase is caused by many factors
  - Hard to predict but fraction of DO bits in the queries is an important factor
- CPU impact is small, Memory impact can be calculated
- Don't add DNSKEY RR set in additional

Questions?