DNSSEC

Tests of Consumer Broadband Routers

Joakim Åhlund & Patrik Wallström, February 2008

.se
Sveriges bästa adress

List of contents

List of figures

1 Introduction

1.1.1 THIS DOCUMENT

This document is a report on the result of performing tests on different kind of broadband routers supplied from different vendors in order to assess the scale of problems with DNSSEC in this environment.

1.2 Abbreviations & Definition of words

Abbrev	*Abbreviation*, this is an abbreviation

1.3 References

[1] References to other documents etc.

1.4 Font

In this document we use the following fonts:

Small bold style	Used for library structure, file names and in- and out puts.
BLOCK LETTERS	Computer names are always written with block letters.

1.5 About .SE

.SE (The Internet Infrastructure Foundation) is responsible for the Internet top-level domain for Sweden. As the central registry, .SE manages domain name registrations and the administrative and technical operation of the national domain name system for .SE.

.SE is an independent non-profit organisation, supporting the positive development of the Internet in Sweden. Through .SE's Internet Fund, the Foundation annually donates means to projects supporting the development and utilisation of the Internet.

For more information, please see: http://www.iis.se/lang/?id=en

2 Introduction - Router tests

2.1 Background

In September 2007 the DNSSEC signed domain gavle.se was hit by availability problems. It was discovered when internet users started to complain.

It was realized that the problems were a combination of certain consumer broadband routers and a bug in the BIND software. The name servers that TeliaSonera and Tele2 used was BIND version 9.4.1, and we discovered that version set the AD bit in the DNS protocol in an unexpected way. What happened was that certain routers didn't let DNS traffic through for DNSSEC signed domains when querying those resolvers. Gavle.se happened to be the first larger domain that normally attracts a lot of ordinary internet users, so this became a significant test of DNSSEC towards those customers, which is something that has not happened before.

The problems that occurred were a result of a bug in BIND *in combination* with some broadband consumer routers. We realized that we needed to perform proper tests of these kind of routers from different vendors in order to assess the scale of the problems of this nature.

2.2 Included and excluded

The test specification is included in appendix 1, section 4.1.

Included in the tests are as follows:

- DNSSEC, with validation both at the resolver and in the client.
- EDNS0
- DNS queries regarding some specific RR types (*ie.* AAAA)
- Open Recursive Resolver
- AS112 queries

Excluded from the tests are:

- Underlying protocols
- Fragmentation of IP packets

There might be other flaws in the broadband consumer routers such as lack of functionality in firewalls or DHCP, but we have limited our tests to DNS.

2.3 How

The tested routers are bought at the nearest and cheapest computer stores, in the same way a regular consumer buys their products.

After the acquisition, the routers have not been upgraded to the latest firmware (if available). The reason is that we don't believe this is something the regular internet user does in general. This behaviour is confirmed by the vendors we have been in touch with.

Our tests are performed using two separate computers. One is placed behind the tested router and is doing the DNS queries, and one is between the ADSL modem and the router sniffing the DNS traffic. The computer doing the sniffing is connected to the net using a wiretap and does not affect the tests.

In order to perform the tests with a setup similar as to what home users are using we are using an ADSL connection. The available ADSL connections are from the two ISP's which at the time when the problems with DNSSEC arose, provided its customers with DNSSEC enabled resolvers. The test queries are sent to their resolvers, which at the time of our tests were running BIND 9.4.2.

For those test queries that tests the bug related to the AD bit and BIND 9.4.1, we have changed the router configuration and set the DNS to use our own 9.4.1 resolvers. Other network configuration has all been setup with DHCP from the ISP.

2.4 Test environment

Our test environment is as the diagram shows below:

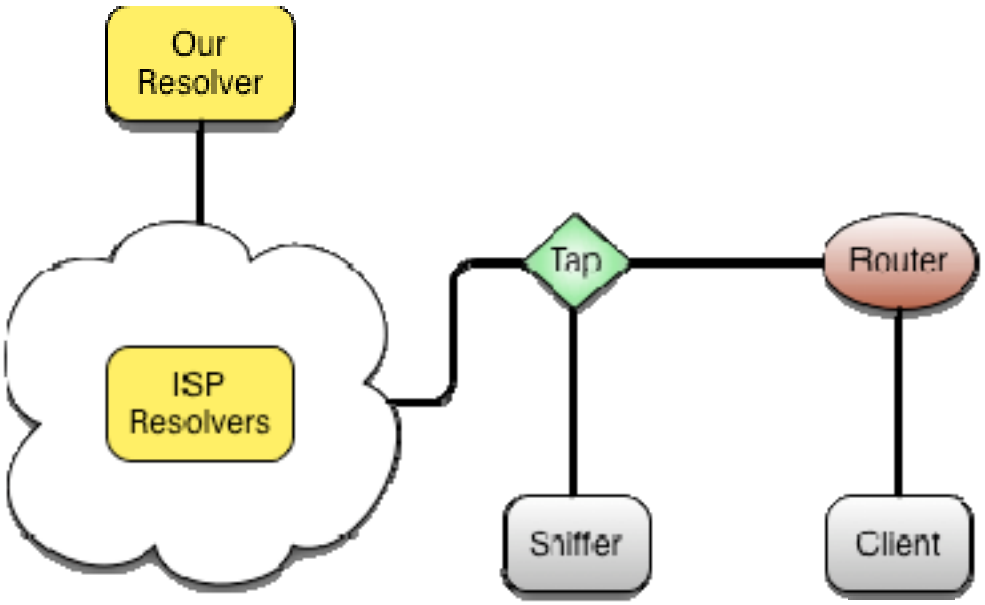

Figure 1: Test environment

The client is an Apple Mac Mini running MacOS X 10.5.1 and dig version 9.4.1-P1.

The router is the tested subject.

Between the router and the internet is a NetOptics Teeny Tap wiretap.

To the wiretap another Mac Mini is connected, running MacOS X 10.5.1. To analyze the traffic we use Wireshark version 0.99.6.

The ISP resolvers are running BIND 9.4.2, and our resolver is running BIND 9.4.1.

2.5 Description of tests

The test specification is included in appendix 1, section 4.1.

The following is a description of what is tested in each query. All tests are run using both UDP and TCP:

The queries in the A section tests answers with large packets in DNS. The answers with each query gradually increases and is tested with increased buffer sizes.

To pass the test all data in the answer must be passed on to the client. Truncated packets and timeouts were two common errors.

Test *B.1* tests if the router forwards the DNS packet if the AD bit is set in the reply, even though validation has not been requested in the query. This is done by querying out BIND 9.4.1 resolver (which has the AD bug) for DNSSEC signed domain. To pass the test the router must forward the packet and the AD bit is set.

In test *B.2* the same query is done as in *B.1* but to the ISP's AD bug fixed resolver. This time the answer is missing the AD bit. To pass this test the router must forward the answer to the client.

In test *C.1* the client queries a DNSSEC signed domain with the DO bit set. To pass the test the answer must include all DNSSEC data from the domain, and that the AD bit is set.

In test *C.2* the client queries a non DNSSEC enabled domain from the ISP resolver with the DO bit set. To pass the test router must forward the answer to the client.

D.1 and *D.2* are almost the same as tests *C.1* and *C.2* but with the addition that they set the CD bit (the resolver should not validate the answer) in the query. This is to test if the router forwards the CD bit untouched. To pass the test the conditions are the same as the C tests but with the difference that the CD bit is set while the AD bit is not.

In test *E.1* the client query our resolver with BIND 9.4.1 after a DNSSEC signed domain with the AD bit set. To pass the test the AD bit must also be set in the answer.

E.2 does the same thing as *E.1* but queries the ISP resolver with BIND version 9.4.2. In this query the answer is expected with an unset AD bit.

F.1 is a test to examine if the router is an open recursive resolver on the routers external interface (the WAN interface). This test is performed on a client computer placed on the Internet. To pass this test the client computer must not receive a working answer.

G.1 queries an IPv6 record in DNS (AAAA).

G.2 queries an SSH fingerprint in DNS (SSHFP).

G.3 queries SRV records.

G.4 queries NAPTR records.

G.5 tests if the router passes on reverse DNS queries about AS112 networks.

3 Summarized results

3.1 Test results

Ten out of twelve routers have passed the tests with mixed results. The two routers we have not managed to test have had such problems that the test results have been useless.

Three of the ten routers have passed the test specification without any remarks. The other seven have had severe problems when using what might be considered plain DNS (no large packets and so on).

The most common errors have been queries and answers over TCP, problems with the AD bit set in the answer and when the client wants to validate DNSSEC (the DO bit set in the query).

The results of our tests are discouraging. What is remarkable is the number of routers that does not handle DNS queries over TCP. But the big problem for DNSSEC is that the majority of the routers don't manage to pass on DNSSEC to the client. This is not a problem as long as the DNSSEC validation is handled by the ISP resolvers, but when there are applications on the client which wants to handle its own DNSSEC validation, it won't work at all in most cases.

The fact that a router does not handle queries over TCP is not very good, especially when there also is a lack of support for EDNS0.

Regarding the last test, *G.5* (AS112), only one router filtered out such queries.

3.2 Vendor reactions

We have contacted all vendors of the tested routers. Some has returned to us with a varying degree of interest. But some has taken the problems seriously and fixed them.

3.3 Continued work

The DNS and DNSSEC problems that we have in the broadband consumer routers are also related to the deployment of IPv6, since the size of DNS packets are expected to grow in time with the deployment of both DNSSEC and IPv6. To fix the routers so that can handle DNS correctly is very important to not limit the growth of these techniques. Internet is today mostly users using these types of products. The next generation Internet should handle both DNSSEC and IPv6 in both networks and DNS with all its applications.

We urge the vendors to take these problems more seriously, and we expect further testing of broadband consumer routers in the future. An international cooperation for these tests should be initiated with a joint web site publishing the specifications as well as the results of all different tests.

4 Appendices

4.1 Appendix 1, DNS test protocol for SOHO-routers

All tests are performed using both UDP and TCP in the query.

****

*** Is the router capable of EDNS0

****

*** Does the router give the client ENDS0 traffic

```
A.1.1: dig +retry=0 +bufsize=512 +qr small.nxdomain.se TXT
A.1.2: dig +retry=0 +bufsize=512 +qr medium.nxdomain.se TXT
A.1.3: dig +retry=0 +bufsize=512 +qr large.nxdomain.se TXT
A.1.4: dig +retry=0 +bufsize=512 +qr huge.nxdomain.se TXT

A.2.1: dig +retry=0 +bufsize=1024 +qr small.nxdomain.se TXT
A.2.2: dig +retry=0 +bufsize=1024 +qr medium.nxdomain.se TXT
A.2.3: dig +retry=0 +bufsize=1024 +qr large.nxdomain.se TXT
A.2.4: dig +retry=0 +bufsize=1024 +qr huge.nxdomain.se TXT

A.3.1: dig +retry=0 +bufsize=4096 +qr small.nxdomain.se TXT
A.3.2: dig +retry=0 +bufsize=4096 +qr medium.nxdomain.se TXT
A.3.3: dig +retry=0 +bufsize=4096 +qr large.nxdomain.se TXT
A.3.4: dig +retry=0 +bufsize=4096 +qr huge.nxdomain.se TXT

A.4.1: dig +retry=0 +bufsize=8192 +qr small.nxdomain.se TXT
A.4.2: dig +retry=0 +bufsize=8192 +qr medium.nxdomain.se TXT
A.4.3: dig +retry=0 +bufsize=8192 +qr large.nxdomain.se TXT
A.4.4: dig +retry=0 +bufsize=8192 +qr huge.nxdomain.se TXT
```

****

**** AD=1 in the reply

****

*** Does the router accept replies with AD=1

```
B.1: dig +retry=0 @validator-with-BIND_9.4.1 +qr dnssec.se SOA
```

*** Does the router accept replies with AD=0

```
B.2: dig +retry=0 @validator-with-BIND_9.4.2 +qr dnssec.se SOA
```

```
************************************************************
```

**** DO=1 in query

```
************************************************************
```

*** Does the router accept queries with DO=1, replies with AD=1

```
C.1: dig +retry=0 @validator-with-BIND_9.4.2 +qr +dnssec dnssec.se SOA
```

*** Does the router accept queries with DO=1, replies with AD=0

```
C.2: dig +retry=0 @validator-with-BIND_9.4.2 +qr +dnssec iis.se SOA
```

```
************************************************************
```

**** DO=1, CD=1 in query

```
************************************************************
```

*** Does the router accept queries with DO=1, CD=1

```
D.1: dig +retry=0 @validator-with-BIND_9.4.2 +qr +dnssec +cdflag dnssec.se SOA
```

*** Does the router accept queries with DO=1, CD=1

```
D.2: dig +retry=0 @validator-with-BIND_9.4.2 +qr +dnssec +cdflag iis.se SOA
```

```
************************************************************
```

**** AD=1 in query

```
************************************************************
```

*** Does the router accept queries with AD=1, replies with AD=1

```
E.1: dig +retry=0 @validator-with-BIND_9.4.1 +qr +adflag dnssec.se SOA
```

*** Does the router accept queries with AD=1, replies with AD=0

```
E.2: dig +retry=0 @validator-with-BIND_9.4.2 +qr +adflag dnssec.se SOA
```

```
************************************************************
```

**** Open resolver in the router? (test from the "WAN side")

```
************************************************************
F.1: dig +retry=0 @router nonexisting.dnssec.se TXT
```

```
************************************************************
```

**** Misc RR types

```
************************************************************
```

*** Does the router let miscellaneous RR types through

```
G.1: dig +retry=0 boa.blipp.com AAAA
G.2: dig +retry=0 boa.blipp.com SSHFP
G.3: dig +retry=0 _sip._tcp.blipp.com SRV
G.4: dig +retry=0 blipp.com NAPTR
```

*** Does the router forward AS112 in-addr.arpa queries

```
G.5 dig +retry=0 1.0.168.192.in-addr.arpa. PTR
```

4.2 Appendix 2, Results from the test protocol

Router:	D-Link DIR-100		D-Link DI-804HV		D-Link DI-624+		Netgear RP614	
Firmware:	v1.00		V1.44		V1.23		V0.1.8_03.17	
Test:	UDP	TCP	UDP	TCP	UDP	TCP	UDP	TCP
A.1.1	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
A.1.2	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.1.3	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.1.4	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.2.1	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
A.2.2	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.2.3	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.2.4	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.3.1	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
A.3.2	FAILED	FAILED	OK	OK	OK	FAILED	FAILED	FAILED
A.3.3	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.3.4	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.4.1	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
A.4.2	FAILED	FAILED	OK	OK	OK	FAILED	FAILED	FAILED
A.4.3	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
A.4.4	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
B.1	FAILED	FAILED	OK	OK	OK	FAILED	FAILED	FAILED
B.2	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
C.1	FAILED	FAILED	OK	OK	FAILED	FAILED	OK	FAILED
C.2	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
D.1	FAILED	FAILED	OK	OK	FAILED	FAILED	FAILED	FAILED
D.2	FAILED	FAILED	OK	OK	OK	FAILED	FAILED	FAILED
E.1	FAILED	FAILED	OK	OK	OK	FAILED	FAILED	FAILED
E.2	FAILED	FAILED	OK	OK	OK	FAILED	FAILED	FAILED
F.1	FAILED	FAILED	OK	OK	OK	OK	FAILED	FAILED
G.1	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
G.2	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
G.3	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
G.4	OK	FAILED	OK	OK	OK	FAILED	OK	FAILED
G.5	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED

Router:	Netgear WNR834B		Netgear WGR614		Netgear WPN824		Linksys WRT54GS	
Firmware:	V1.0.4.0WW		V2.0.20_1.0.20		V2.0.10_1.2.17		v1.50.6	
Test:	UDP	TCP	UDP	TCP	UDP	TCP	UDP	TCP
A.1.1	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
A.1.2	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.1.3	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.1.4	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.2.1	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
A.2.2	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.2.3	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.2.4	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.3.1	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
A.3.2	FAILED	FAILED	OK	FAILED	OK	FAILED	OK	OK
A.3.3	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.3.4	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
A.4.1	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
A.4.2	FAILED	FAILED	OK	FAILED	OK	FAILED	OK	OK
A.4.3	FAILED	FAILED	OK	FAILED	FAILED	FAILED	OK	OK
A.4.4	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	OK	OK
B.1	FAILED	FAILED	OK	FAILED	Untested	Untested	OK	OK
B.2	FAILED	FAILED	OK	FAILED	OK	FAILED	OK	OK
C.1	FAILED	FAILED	OK	FAILED	OK	FAILED	OK	OK
C.2	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
D.1	FAILED	FAILED	OK	FAILED	OK	FAILED	OK	OK
D.2	FAILED	FAILED	OK	FAILED	OK	FAILED	OK	OK
E.1	FAILED	FAILED	OK	FAILED	Untested	Untested	OK	OK
E.2	FAILED	FAILED	OK	FAILED	OK	FAILED	OK	OK
F.1	OK	OK	OK	OK	OK	OK	OK	OK
G.1	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
G.2	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
G.3	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
G.4	OK	FAILED	OK	FAILED	OK	FAILED	OK	OK
G.5	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED	FAILED

Router:	Zyxel P-320W		FON	
Firmware:	V1.00(ZH.3)C0		0.7.2 r2	
Test:	UDP	TCP	UDP	TCP
A.1.1	**OK**	**FAILED**	**OK**	**OK**
A.1.2	**FAILED**	**FAILED**	**OK**	**OK**
A.1.3	**FAILED**	**FAILED**	**OK**	**OK**
A.1.4	**FAILED**	**FAILED**	**OK**	**OK**
A.2.1	**OK**	**FAILED**	**OK**	**OK**
A.2.2	**FAILED**	**FAILED**	**OK**	**OK**
A.2.3	**FAILED**	**FAILED**	**OK**	**OK**
A.2.4	**FAILED**	**FAILED**	**OK**	**OK**
A.3.1	**OK**	**FAILED**	**OK**	**OK**
A.3.2	**FAILED**	**FAILED**	**OK**	**OK**
A.3.3	**FAILED**	**FAILED**	**OK**	**OK**
A.3.4	**FAILED**	**FAILED**	**OK**	**OK**
A.4.1	**OK**	**FAILED**	**OK**	**OK**
A.4.2	**FAILED**	**FAILED**	**OK**	**OK**
A.4.3	**FAILED**	**FAILED**	**OK**	**OK**
A.4.4	**FAILED**	**FAILED**	**OK**	**OK**
B.1	**OK**	**FAILED**	**Untested**	**Untested**
B.2	**OK**	**FAILED**	**OK**	**OK**
C.1	**FAILED**	**FAILED**	**OK**	**OK**
C.2	**OK**	**FAILED**	**OK**	**OK**
D.1	**FAILED**	**FAILED**	**OK**	**OK**
D.2	**FAILED**	**FAILED**	**OK**	**OK**
E.1	**OK**	**FAILED**	**Untested**	**Untested**
E.2	**OK**	**FAILED**	**OK**	**OK**
F.1	**OK**	**OK**	**OK**	**OK**
G.1	**OK**	**FAILED**	**OK**	**OK**
G.2	**OK**	**FAILED**	**OK**	**OK**
G.3	**OK**	**FAILED**	**OK**	**OK**
G.4	**OK**	**FAILED**	**OK**	**OK**
G.5	**FAILED**	**FAILED**	**OK**	**OK**