The shiny new cryptographically secured globally distributed database

Presented by
Paul Wouters
seceng, Red Hat Inc.
Topics

- DNSSEC theory in 7 screen shots
- DNSSEC software: validating, signing
- Converting applications to use DNSSEC
- Using DNSSEC for non-DNS purposes
  - TLSA, SSHFP, IPSECKEY, <your crazy idea here>
DNSSEC in 7 screen shots
Image a DNS RRset

[paul@thinkpad ~]$ dig fedoraproject.org

; <<>> DiG 9.9.1-P2-RedHat-9.9.1-5.P2.fc17 <<>> fedoraproject.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61882
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
fedoraproject.org. IN A

;; ANSWER SECTION:
fedoraproject.org. 44 IN A 209.132.181.16
fedoraproject.org. 44 IN A 85.236.55.6

;; Query time: 95 msec
;; SERVER: 193.110.157.123#53(193.110.157.123)
;; MSG SIZE  rcvd: 78

[paul@thinkpad ~]$
Add DNS signature record

[paul@thinkpad ~]$ dig +dnssec fedoraproject.org

; <<>> DiG 9.9.1-P2-RedHat-9.9.1-5.P2.fc17 <<>> +dnssec fedoraproject.org
; ; global options: +cmd
; ; Got answer:
; ; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 206
; ; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

; ; OPT PSEUDOSECTION:
; ; EDNS: version: 0, flags: do; udp: 4096
; ; QUESTION SECTION:
; ;fedoraproject.org.

; ; ANSWER SECTION:
fedoraproject.org. 60 IN A 209.132.181.16
fedoraproject.org. 60 IN A 85.236.55.6
fedoraproject.org. 60 IN RRSIG  A 5 2 60 20120923193204 20120824193204 7725 fedoraproject.org. s8b4b1bxfiQwis6xh8fv+dnulvgoHmi//czo6G0CGye2ffSoX9ibhd4zU UWfdchCTuoUYOJGCqYgVbLYGZhN4JeVua0IoX7hBz3ISxR/FqihtsDf+Q/TQ2yu30DnWssRQUPRfclXVU8adB+utsXL3FYAhTSDyf/GezjTgUQXq080=

; ; Query time: 201 msec
; ; SERVER: 193.110.157.123#53(193.110.157.123)
; ; WHEN: Sat Aug 25 18:46:59 2012
; ; MSG SIZE  rcvd: 255

[paul@thinkpad ~]$
Also signature for NXDOMAIN
Publish the public key used in DNS

[paul@thinkpad ~]$ dig +dnssec -t dnskey fedoraproject.org

; <<<< DiG 9.9.1-P2-RedHat-9.9.1-5.P2.fc17 <<>> +dnssec -t dnskey fedoraproject.org
; global options: +cmd
; Got answer:
; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 47954
; flags: qr rd ra ad; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1

; ANSWER SECTION:
fedoraproject.org. 257 3 5 AwEAAAdTXJc0joiKGfTvLXi+Lx6pKvPv0oJESt9PR8TCCxGVv7h3BY3 uXLkJckuTQ0aOCp2KF8zHgNgpMK03p1fd94pn9JZuxfKvKsiYH2KvN0 a/6550Pj06jRhqAP5grX01Iz48H411ZhGxI01Bz7t01wAa0zj MJLUG ChRj8gV+3L0q66T8z1QRF33t9UMH5REaSFaqf/ztssfJdYktGoZi3 nFW7A745+QbM1LWXOwq3FcYPVzhH0807/7WpxmzMG/ET8VeqIsvh8E nZNDNNfJyPbY9B1OIrFCPe03ALgFNemaBZwmeQaX+D4Duup5xG0mdtC 046SpM1YH6c=
fedoraproject.org.

IN  DNSKEY 256 3 5 AwEAAAcWNOWl5pCI3100P2r8nStL60Zjb/2JQ0lytamVapOL44z0YWf t 7pu0hx3cnIMlaj00sEwbg2/10lYy+38cYqDXb5dFg1I6Gz10S5xN7zr 9hz5RSK5N2jkycdj/B0Byi34V+XGpDqfG4I97+8sIzSr60TmGAKTtv9v iL3ByeCN
fedoraproject.org.

IN  RRSIG  DNSKEY 5 2 300 20120923193204 20120824193204 7725 fedoraproject.org.

g. ZTeiBc_04w5px0g65qDxa8P1xUDnSdIQ:1jI1CrP0LA:RLpB51eUL0n_1Dpe2axXW2N78fAfpF+PocRURS1o6Q55GtGgd060nPUENLC 0Uyj1sVPZ Z1TVV+nu4RdL4yIxeOEh25t0DXVe0qngne9w6+1i/Hg9ITNzTLjyB8p bHY=
fedoraproject.org.

IN  RRSIG  DNSKEY 5 2 300 20120923193204 20120824193204 16207 fedoraproject.org.

g. U15SP5b6eo/0b9TYffBcnTCLhdtddy69LFVEe0PEFUQ/6mtykt.5NhkJ9j7x3Zk35vsAT/fyAvVn9elsIXk/G2NR22/2mmAcvfoID8 9J/EXDcGCh A17q70j8LSkemMxsMy7ek4yLd83s2+00wonnaitsIS4sE60jcz6M00L9v9uh UzjfpMFouB0eqTEjwBHDiQUkij40rLG7AzmL/ t:9xAmTRPwJC4h2k2J wYMEiCrlab6MNJAJAzrBximJPPeYzi96g4WzFnX1QFpaKFz5noV7Af9gFg EUtmt7Z7vHcc1u/ryY+Oc9XvakanjG V0Lrg6nJiFACu1F5qNgNvzGAkY 8dL+rg==
Hash of public key goes to parent
Build DS -> DNSKEY trust chains

DNSSEC Trust tree:
dnssec.se. (A)
    |-- Existence is denied by:
    |   |-- dnssec.se. (NSEC _adsp._domainkey.dnssec.se. NS SOA TXT RRSIG NSEC DNSKEY SPF)
          |-- dnssec.se. (DNSKEY keytag: 30332 alg: 5 flags: 256)
          |   |-- dnssec.se. (DNSKEY keytag: 2467 alg: 5 flags: 257)
          |   |-- dnssec.se. (DS keytag: 2467 digest type: 1)
          |   |   |   |-- se. (DNSKEY keytag: 12318 alg: 5 flags: 256)
          |   |   |   |   |-- se. (DNSKEY keytag: 59747 alg: 5 flags: 257)
          |   |   |   |   |-- se. (DS keytag: 59747 digest type: 2)
          |   |   |   |   |   |   |--. (DNSKEY keytag: 50398 alg: 8 flags: 256)
          |   |   |   |   |   |   |   |--. (DNSKEY keytag: 19036 alg: 8 flags: 257)
          |   |   |-- dnssec.se. (DS keytag: 2467 digest type: 2)
          |   |   |   |   |-- se. (DNSKEY keytag: 12318 alg: 5 flags: 256)
          |   |   |   |   |   |-- se. (DNSKEY keytag: 59747 alg: 5 flags: 257)
          |   |   |   |   |   |-- se. (DS keytag: 59747 digest type: 2)
          |   |   |   |   |   |   |--. (DNSKEY keytag: 50398 alg: 8 flags: 256)
          |   |   |   |   |   |   |   |--. (DNSKEY keytag: 19036 alg: 8 flags: 257)
          |   |   |   |-- se. (DNSKEY keytag: 50398 alg: 5 flags: 257)
          |   |   |   |   |-- se. (DS keytag: 50398 digest type: 2)
          |   |   |   |   |   |   |   |--. (DNSKEY keytag: 19036 alg: 8 flags: 257)
Existence denied
    |   |-- Chase successful
[...]
[paul@thinkpad ~]$
DNSSEC Lookaside Verification

$paul@thinkpad:~$ dig +dnssec -t dlv fedoraproject.org.dlv.isc.org

; <<>> DiG 9.9.1-P2-RedHat-9.9.1-5.P2.fc17 <<>> +dnssec -t dlv fedoraproject.org.dlv.isc.org
; global options: +cmd
; Got answer:
; -->HEADER<<- opcode: QUERY, status: NOERROR, id: 54192
; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 5, ADDITIONAL: 1

; QUESTION SECTION:
fedoraproject.org.dlv.isc.org. IN DLV

; ANSWER SECTION:
fedoraproject.org.dlv.isc.org. IN DLV 16207 5 1 8DD099791A2A110851FDE5D14F6C62ADC3DD7C18 fedoraproject.org.dlv.isc.org. IN DLV 16207 5 2 A7C9BF5AFE374C9650ED678F3D38931A7DE9256B86A7BC34 D6DEED7D 4E492E5E fedoraproject.org.dlv.isc.org. IN RRSIG DLV 5 5 3600 20120924203004 20120825203004 64263 dlv.isc.org. gDSRB0ybiCr346RPL71B01rE6CMrAp0cbbKHTJEUaF1tu13R9vgWn p9+ll7CwzMZf5E8KJluTA9ShRxPr3X9vhbyszMM6CvW4Fz Wc9iBAHwBR ScVJBYpsi4hqwGh8Xc8uX/rzkKBg7Fqa+R7qWshHMUuEJnracS0/jkMzP_Epc=

; AUTHORITY SECTION:
dlv.isc.org. 2610 IN NS ns.isc.afilias-nst.info.
dlv.isc.org. 2610 IN RRSIG NS 5 3 3600 20120924203004 20120825203004 64263 dlv.isc.org. gA20V4NhDFzJSYd83TEJb1pq4ef0GL70CMvJtizAvmBcqFSMXEccZDYF IHicKFDFwFrFeJAO2/9MYpdVi9Ic0JvinsxY7mEWEcwr2N2 sLV0vK74mA DQHQMx0aNhHxqupFWSrBq3hPhe5H0Atd9HjHfVKBKKEQaDkcAwEDARg h/A=

[paul@thinkpad ~]$
DNSSEC states and bits

- Secure: validated from known trust anchor key
- Insecure: proven no trust anchor exists there
- Bogus: crypto failed, answer scrubbed (ServFail)
- Indeterminate: answers incomplete/missing

- Query using “dig +dnssec”
- Check dig output for “AD” - Authenticated Data
- Debug ServFail's using “dig +cd +dnssec”
DNSSEC in Linux distro's

- DNSSEC capable DNS resolvers
  - unbound (preferred for on the fly reconfiguration)
  - bind (named)
- DNSSEC capable DNS servers
  - All modern DNS servers (bind, nsd, powerdns)
- DNSSEC zone signers
  - opendnssec, dnssec-signzone (bind), pdns, dnssec-tools, ....
- DNSSEC utilities (dig, unbound-host, drill,..)
  - yum/apt-cache search dnssec
DNSSEC validation in Fedora / RHEL

- yum install unbound or yum install bind
- echo "nameserver 127.0.0.1" > /etc/resolv.conf
- No further configuration needed, DNSSEC enabled in default configuration since Fedora 15

- Don't actually do this on your laptop, as you depend on spoofed DNS every day!
DNSSEC resolving issues

- DNSSEC too good – protects against
  - hotspot / captive portal
  - VPN – private views
  - opendns, NXDOMAIN squatting, dns rewriting
- Many applications mess with /etc/resolv.conf
- We need to address these issues all at once
DNSSEC and hotspots

- NetworkManager, unbound, dnssec-triggerd
- Run DNSSEC server locally: unbound
- dnssec-triggerd with NM hook to:
  - Detect hotspot via http://fp.org/static/hotspot.txt
  - use resolv.conf to temporarily bypass unbound
  - Launch browser to hotspot-nocache.fp.org
  - Detect payment / license agreement
  - Re-enable DNSSEC using unbound via resolv.conf
Hotspot detected

Web traffic hijacked

The web traffic on this network is being hijacked. Is this a hotspot?

While you log in you are insecure, until the traffic hijacking has stopped and dnssec-trigger has detected regular web access.

Skip if you are not logging into a hotspot right now.

Skip  Log in
Login to hotspot
Fallback to DNS over TCP

probe dnssec results

dnssec-trigger 0.11
results from probe at 2012-08-25 16:51:55

ssl443 80.239.156.220: OK
tcp80 152.19.134.150: OK
authority 192.33.4.12: error timeout
http fedoraproject.org (209.132.181.16): OK
cache 192.168.101.1: error timeout

DNSSEC results fetched from open resolvers over TCP

Repprobe
Probe results
Hotspot signon
Quit
Or worse: cache-only

The Network Fails to Support DNSSEC

The network you are connected to does not allow DNSSEC, via the provided DNS caches, nor via contacting servers on the internet directly (it filters traffic to this end). It is not possible to provide DNSSEC security, but you can connect insecurely.

Do you want to connect insecurely?

* if you choose **Disconnect** then DNS is disabled. It is safe, but there is very little that works.

* if you choose **Insecure** then the DNSSEC security is lost. You can connect and work. But there is no safety. The network interferes with DNSSEC, it may also interfere with other things. Have caution and work with sensitive personal and financial things some other time.

Some hotspots may work after you have gained access via its signon page. Then use **Reprobe** from the menu to retry.

*Stay safe out there!*
DNSSEC and VPNs

- Openswan reconfigures unbound on the fly

  - IPsec server sends XAUTH domain name and name server parameters to openswan client (i.e. “redhat.com”, 10.11.255.156)

  - Openswan informs unbound: “unbound-control forward_add redhat.com 10.11.255.156”

  - On termination, openswan issues “unbound-control forward_remove redhat.com” and “unbound-control flush_requestlist”
DNSSEC zone signing

- yum install opendnssec -y
- systemctl ods-enforcerd start
- systemctl ods-signerd start
- ods-ksmutil zone --add yourzone.com --input /var/named/yourzone.com --output /var/named/yourzone.com.signed
- ods-signer sign yourzone.zome
  (updated named.conf, restart named, wait a few days, go to Registrar for DS, or dlv.isc.org to publish DLV)
- ods-ksmutil key ds-seen --zone yourzone.com \ --keytag xxxxx
Convert code to use DNSSEC

- We will use libunbound as our API
- Find gethostbyname() calls (direct / indirect)
- Initialize a DNSSEC cache context
- Configure its behaviour to emulate POSIX
- Load DNSSEC trust anchor keys (root, DLV)
- Call ub_resolv() directly or via thread / callback
- Check return value for DNSSEC parameters
/* Converting gethostbyname() to libunbound with DNSSEC support */

#include <unbound.h>
struct ub_ctx* dnsctx;

int unbound_init(int verbose)
{
    dnsctx = ub_ctx_create(); /* create unbound resolver context */

    if(verbose) {
        printf("unbound context created - setting debug level high\n");
        ub_ctx_debuglevel(dnsctx,255);
    }

    /* look at /etc/hosts before DNS lookups as people expect this */

    if( (ugh=ub_ctx_hosts(dnsctx, "/etc/hosts")) != 0) {
        printf("error reading hosts: %s. errno says: %s\n", 
            ub_strerror(ugh), strerror(errno));
        return 0;
    }

    /* Use DHCP obtained DNS servers as forwarding cache */

    if( (e = ub_ctx_resolvconf(dnsctx, "/etc/resolv.conf")) != 0) {
        printf("error reading resolv.conf: %s. errno says: %s\n", 
            ub_strerror(e), strerror(errno));
        return 0;
    }

    ....

"unbound-hooks.txt" 216L, 6252C written
Add trusted DNSSEC keys

/* DNSSEC root key */
static char *rootanchor = "". IN DNSKEY 257 3 8 AwEAAagAIklVzrC6Ia7gEzah0R+9W29euXJhVVL0qSEw008gcCjFFVQUTf6v58fl
jWBdoYIoErAcQqBGczh/RStIo08gONfnfL2MTJRkxoXbfDaUeVPQyuYEhg37NZWJ09VnMVDxP/VHL496H/QZxKjř5/Efucp2gaDX6RSrCxpoY68Lsv
PVjrROZsww31apAzvN9dlzEheX7ICJBBtuA6G3L0pzw5h0A2hzCTMjJPJ8LbqF6dsV6DoB0zgu0sGicG0Yl70y0dXfZ57reLSQgeu+ipAdTTJ25AsR
TAoub80NGclmqrAnRLKBP1dfwhYB4N7knNnulQxA+Uk1ihz0=";

/* DNSSEC DLV key, see http://dlv.isc.org */
static char *dlvanchor = "dlv.isc.org. IN DNSKEY 257 3 5 BEAAAPHMu5/onznrEE7v1egmhg/WP00+juoZrW3euWEhE4MxQcEl+llY2br
hQv5rN32RkTmX6Mj70jdeN4Xknw58dnJNPCxn8+jA6l2FZLK8t+1uq4W+mna3q2+DL+k6BD4mewMLb1YFwe0PG73Te9fZ2kJb56dhgMde5ymX4B
I/oQ+cAK50/xvjiV00FrFr8kw6ucMNcTfwFlgPe+jnGxPPEEmHate/URkY62ZfkLbAADDLH09Ir5ttrAy7mbBZVc0wleU/Rw/mRx/wwNMCTgNbsMQKtuUdvNX
DrYJD5Hws3xiRX1Fr+al59UmZfSav/4NWLkjHzpT59k/VStTDNGYuWBNh";

/* real errno handling code removed for clarity */

/* add trust anchors to libunbound context */
if(verbose)
printf("Loading root key:%s\n", rootanchor);
e = ub_ctx_add_ta(dnsctx, rootanchor);

/* Enable DLV */
if(verbose)
printf("Loading dlv key:%s\n", dlvanchor);
e = ub_ctx_set_option(dnsctx, "dlv-anchor:"+dlvanchor);

return 1; /* real errno handling code removed for clarity */
Add DNSSEC resolve call

```c
#include <stdio.h>
#include <sys/socket.h>
#include <stdlib.h>
#include <arpa/inet.h>
#include <openssl/ssl.h>
#include <openssl/err.h>
#include <openssl/x509.h>

int main(void)
{
    SSL_CTX *ctx;
    char *err = NULL;
    int qtype = 1; /* default to IPv4 */
    int e;
    struct ub_result* result;

    e = ub_resolve(dnsctxt, src, qtype, 1 /* CLASS IN */, &result);
    if(result->bogus) {
        fprintf(stderr,"ERROR: %s failed DNSSEC validation!\n", result->qname);
        return -1;
    }

    if(!result->havedata) {
        if(result->secure)
            printf("Validated reply proves '%s' does not exist\n", src);
        else
            printf("Failed to resolve '%s' (%s)\n", src, (result->bogus) ? "BOGUS" : "insecure");

        ub_resolve_free(result);
        return err;
    } else if(!result->bogus) {
        if(!result->secure) {
            fprintf(stderr,"warning: %s lookup was not protected by DNSSEC!\n", result->qname);
        }
    }
}
```
replace gethostbyname()

/* Code changes to support DNSSEC in openswan's "add connection" code */

#ifdef DNSSEC
    if(resolvip) {
        /* initialise our DNSSEC resolver context */
        if(!unbound_init(verbose)){
            printf(stderr,"unbound_init() failed, aborting\n");
            return 1;
        }
    }
#endif

    if(hostname) {
        err_t e;
        char b[ADDRTOT_BUF];
    
#ifdef DNSSEC
        if( verbose) {
            printf("Calling unbound_resolve() for hostname value");
        }
        e = unbound_resolve(hostname, strlen(hostname), AF_INET, &cfg->dr);
#else
        /* toaddr() calls gethostbyname(hostname) */
        e = ttoaddr(hostname, strlen(hostname), AF_INET, &cfg->dr);
#endif

    
#endif

    ub_ctx_delete(dnsctx);
#endif

exit(exit_status);
}
Achievement unlocked!

- Your zone is continuously signed and updated
- Your resolvers are deployed with DNSSEC
- You can handle necessary spoofed data from VPN and hotspots
- Your application is DNSSEC aware and protects against DNS spoofing and cache poisoning
- You can now use DNSSEC to securely publish your own data
non-DNS data use of DNSSEC

- TLSA – Store HTTPS certificates in DNS
- SSHFP – Store ssh known_hosts keys in DNS
- IPSECKEY – Store IPsec public RSA keys in DNS
- S/MIME – Store email public keys in DNS
- SMTP/TLSA – STARTSSL public keys in DNS

(first three are already described in RFCs, the last two are currently still drafts)
The TLSA record

2.1. TLSA RDATA Wire Format

The RDATA for a TLSA RR consists of a one-octet certificate usage field, a one-octet selector field, a one-octet matching type field, and the certificate association data field.

_443._tcp.fedoraproject.org. 300 IN TLSA 3 0 1 F4BF2EAD76DA47E2EB64D6BD8033 \\ 5B276574E8E62617908D4917F19E 75920F22
Other data suggestions

- PGP/GPG fingerprints in DNS?
- OTR (IM) fingerprints in DNS?
- File hashes in DNS? (rpm, tripwire, IMA/EVM)
- SElinux policies via DNS?
- Software Update Versions in DNS?
- Distributed secure twitter-like publishing?
  1. tweets.fp.org. IN TXT “#dnssec in @fedora is neat!”
  2. tweets.fp.org. IN TXT “#linuxcon people think I'm nuts”
Offline DNSSEC chains

- My laptop stores DNSSEC hierarchy from the root (".")) to itself ("pwouters.redhat.com")
- Your laptop does same, from "." to "johndoe.toronto.example.ca"
- Laptops can now authenticate each other offline via adhoc/bluetooth – no internet required as long as both have the root (".")) key.
DNSSEC and Firefox

- addon: DNSSEC Validator (labs.nic.cz)
- addon: Extended Validator (os3sec.org)
- addon: DNSSEC / TLSA validator
  - people.redhat.com/pwouters/
- All proof of concept addons to push browser vendors for native integration
DNSSEC Validation

Untrusted Connection - Mozilla Firefox

Invalid domain name signature has been detected. It could indicate spoofed connection!

This website does not supply identity information.

Your connection to this website is not encrypted.

Normally, when you try to connect securely, sites will present trusted identification to prove that you are going to the right place. However, this site’s identity can’t be verified.

What Should I Do?

If you usually connect to this site without problems, this error could mean that someone is trying to impersonate the site, and you shouldn’t continue.

Get me out of here!

Technical Details:

TLSA / DNSSEC Validation

Domain name is secured by DNSSEC, and TLSA proved the certificate is valid (and no CA)

You are connected to

nohats.ca

which is run by
(unknown)

Your connection to this website is encrypted to prevent eavesdropping.
Questions?
Ideas?

Contact:
pwouters@redhat.com
LetoAms on FreeNode, Twitter, etc
But djb says 'DNSSEC is evil'

- DNSSEC does not cause 51x amplification (numbers published by Dan Kaminsky and me)
- DNS privacy is more then just encryption
- DNScurve would destroy all DNS caches (causing much worse amplification)
- DNScurve causes CPU load on DNS auth servers (talk about Denial of Service attack)
- The OpenDNS business model is forging dns...
- DJB is wrong – come talk to me afterwards
But Moxie Marlinspike says 'DNSSEC and Verisign are evil'

- 200+ million domain names, can't store/verify
- X-Files was wrong, you need to trust someone
- Hierarchical trust or decentralized trust?

- "Peer to Peer" DNS cannot work, uniqueness requires enforcement, human-readability
- Moxie is postponing the inevitable trust. come talk to me after the presentation