DNSSEC Key Management

Part 1 of 3 – Key Storage

About the participants

- Alan Clegg
  - ISC Training and Support Engineer

    e-mail - **aclegg@isc.org**

    twitter - **@knobee**

- Larissa Shapiro
  - ISC Product Manager

    e-mail - **larissas@isc.org**

    twitter - **@ISCdotORG**

Part One? What's this?

- Other presentations on DNSSEC Key Management will follow, including:
  - Key Rollover Policies
  - Key Rollover Tools and Practices
- But for now, let's talk storage.

What are keys for anyway?

- Zone Signing Key
  - Creates the signatures on the resource record sets in the zone
- Key Signing Key
  - Creates a signature on the DNSKEY resource record set
  - Provides the "Secure Entry Point" into this zone from the parent

How do I create them?

- BIND distribution provides the application `dnssec-keygen`
- Generates both ZSK and KSK
  - Type of keys depends on flag option

    `-f ksk`
- Each run of `dnssec-keygen` creates two different files...

Two files?

- DNSSEC uses Public Key encryption
  - Private and Public portion for each key
  - Private portion must remain secret
  - Public portion is published in zone data

KSK and ZSK into the zone!

- To allow remote validation of the zone data, the public portions of the KSK and ZSK must be included in the zone before signing:

`$INCLUDE` or cut-and-paste

- Automation in BIND 9.7

How does signing happen?

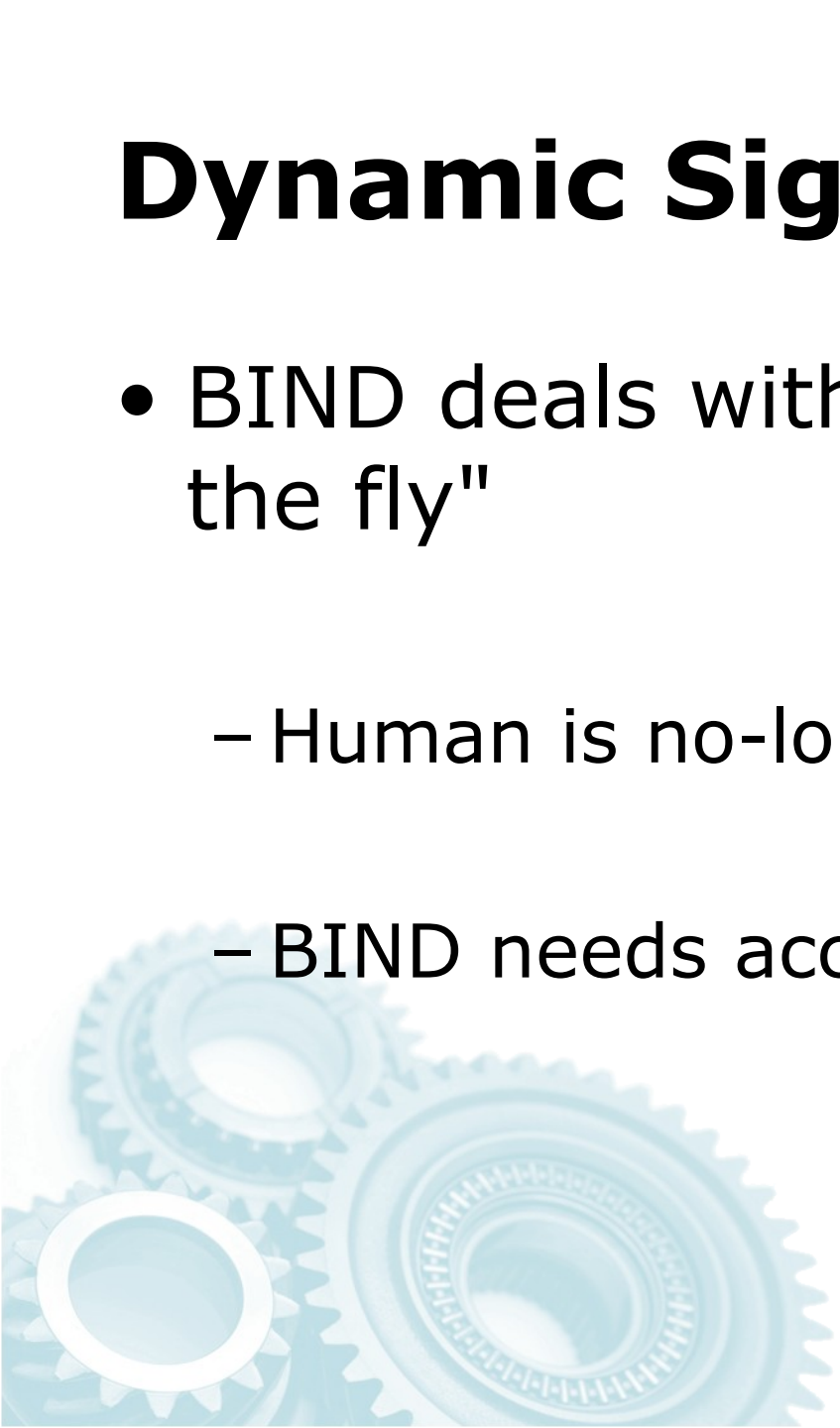
- Prior to BIND 9.7
  - Zones manually signed using "`dnssec-signzone`"
- Since 9.7
  - Manual signing is enhanced
  - Automatic (online) re-signing available

Manual Signing

- Entire signing process is done from the command line
  - Operator must have access to both public (included) and private (signing) portions of the key
  - Unsigned zone is completely signed
  - Already signed zone is re-signed (as needed)

Dynamic Signing

- BIND deals with signing the zone "on the fly"
  - Human is no-longer in the loop
  - BIND needs access to the keys



Where do we store keys?

- Two current options:
  - In the filesystem
  - Hardware Security Module (HSM)

Filesystem – Good enough?

- Prior to 9.7, keep the keys with the zone data was the best choice..

  Pro:

  Simple to tell which keys are available for each zone

  Con:

  Intermingles "private" and "public" data

So, 9.7 makes this better?

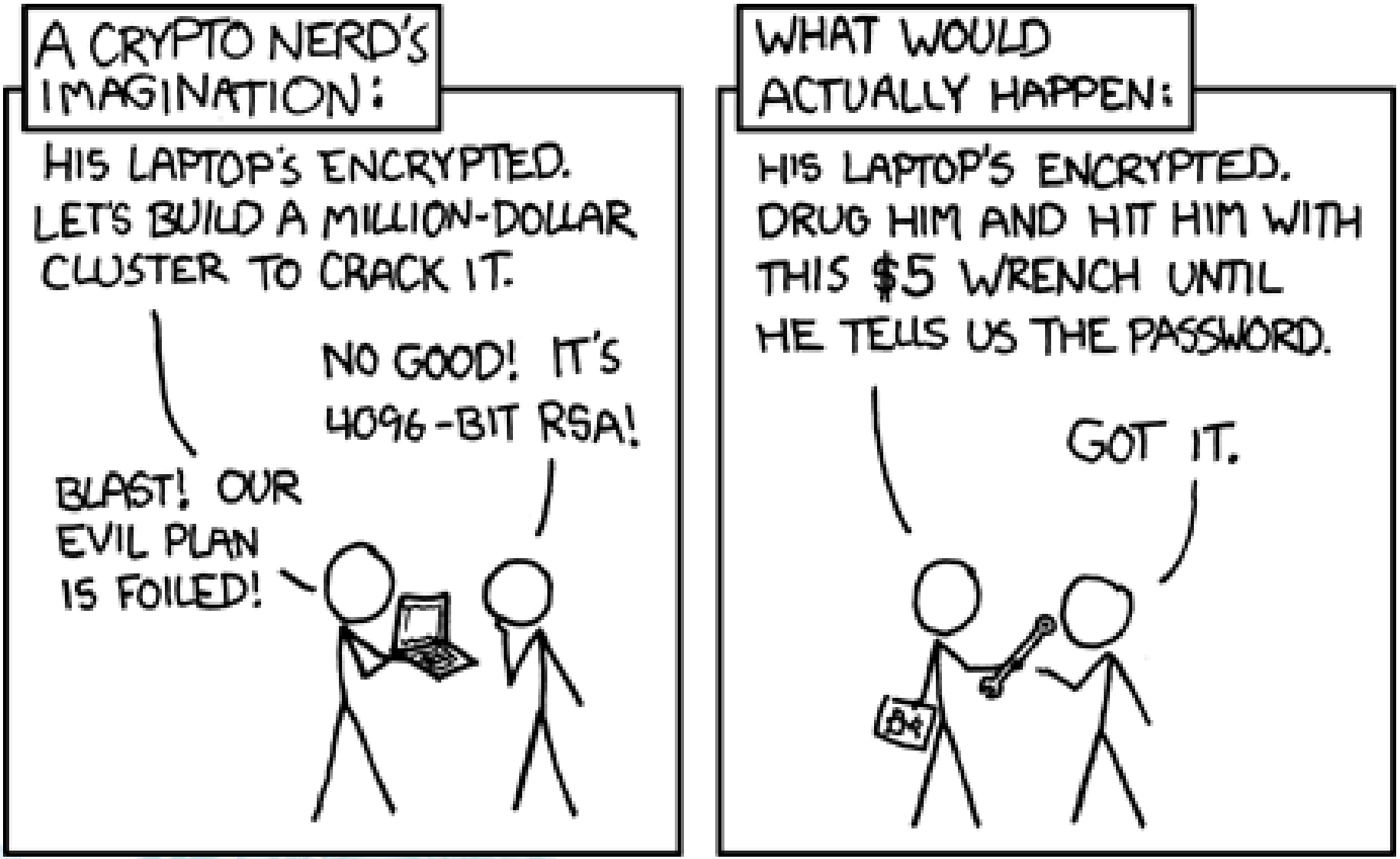
- With 9.7, DNSSEC tools gained a "`-K`" option
  - Specifies location of key directory
    - For signing, read key from there
    - For generating, write key to there

Keys go where?

- In addition, `named` has a matching option

```
key-directory
```

- Configured per-zone or at global scope



http://www.xkcd.com/538/

I'd like to not be beaten..

- KSK private key files can be kept off-line until needed
  - Since the KSK only signs the DNSKEY resource record set, it is only needed during changes of that RRset
  - KSK or ZSK rollovers

Don't the RRSIGs expire?

- The signature on the DNSKEY RRSET can be artificially extended so that it is not re-created every 30 days.
  - This makes the key more vulnerable to replay attacks (the reason for signature expiration)

If I need more security...

- The filesystem is always vulnerable to local "attacks" by privileged users
  - "high value" zones need better protection
  - Mandates are sometimes motivation for better protection (FIPS 140-2)

Hardware Security Module

- Hardware Security Modules (HSMs) provide non-extractable private keys

- Public key portion is still in the filesystem

  - Needed for "inclusion" into the zone

Ok, how does it work?

- Signing application must have access to hardware device
  - `named` or `dnssec-signzone`
  - Access through a modified OpenSSL library

Signing...

- The signing of zone data takes place in the Hardware Security Module
  - The KSK private data is never exposed to a potential attacker
  - RRs signed/second depends more on the HSM

Modified OpenSSL?

- Patches are included in BIND distributions
  - Adds "key by reference" and PIN management
  - Does NOT replace system version of OpenSSL

So, does it play nicely?

- Key creation with an HSM uses "`pkcs11-keygen`" instead of "`dnssec-keygen`"

- Automatic re-signing (via `named`) is able to be automated
  - PIN must be stored in the filesystem...

Where do I get one?

- Keyper devices are available individually and as a bundled consulting/install product from ISC

- Contact your ISC account manager for more information

Questions or comments?