DNS(SEC) Views

https://dnssecviews.net

Pouyan Fotouhi Tehrani, Eric Osterweil,
Thomas C. Schmidt, Matthias Wählisch

Motivation

- Securing DNS zones is fairly straight-forward
- Authoritative nameservers provide consistent data

} We have been monitoring this through SecSpider (https://secspider.net/)

HOWEVER

- Users rely on recursive resolvers
- Recursive resolvers follow different policies
- Timing, caching, multiple signers, etc. influence propagation
- Data from multiple sources may be combined to validate signed recods
- Infrastructure providers are interested to know how their services are observed by users

That's why we built the **DNS(SEC) Views**!

Motivation

- Securing DNS zones is fairly straight-forward
- Authoritative nameservers provide consistent data

We have been monitoring this through SecSpider (https://secspider.net/)

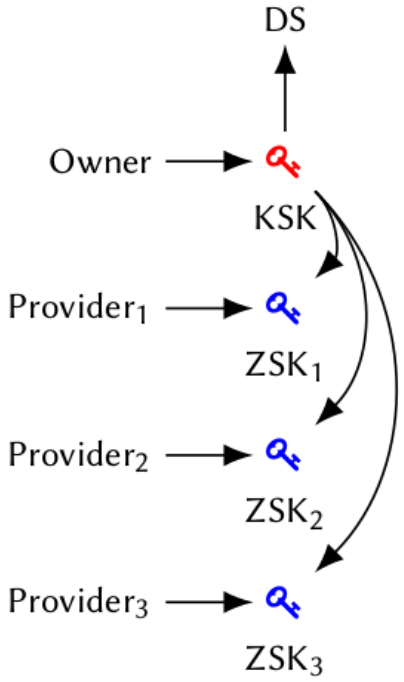
Goal: understand how the **distributed** nature of DNS and its **eventual consistency** (temporal aspect) is observed by and affects users

- Users rely on re
- Recursive resol
- Timing, caching
- Data from multiple sources may be combined to validate signed recods
- Infrastructure providers are interested to know how their services are observed by users

That's why we built the **DNS(SEC) Views**!

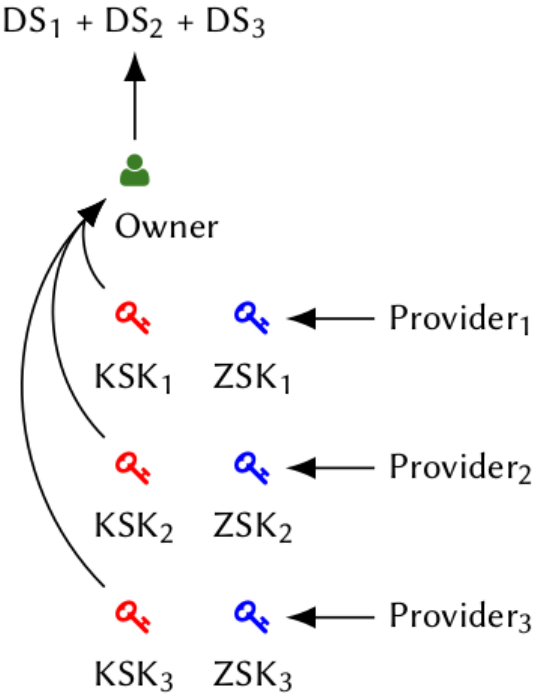
Use Case: Multi-Signer DNSSEC

Common KSK Set, Unique ZSK Set per Provider

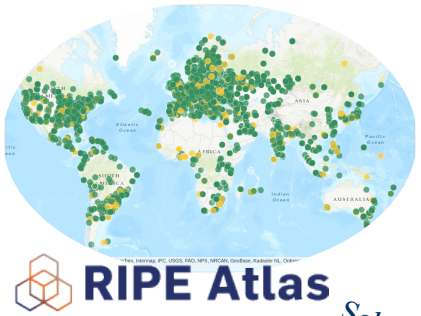


To verify correct deployment observations from various vantage points should simultaneously be collected.

Unique KSK Set and ZSK Set per Provider



System Overview



Database

Persist and query

Schedule Measurements

Listen for results

RESTful API

Front End

Register Zone

Observe Stats

Back End

Infrastructure Operator

Approach: Collect Data

1. Find zone apex
2. Schedule regular measurements via RIPE Atlas for following records:
   - `DNSKEY`
   - `DS`
   - `NS`
   - `SOA`
3. Parse and serialize data into the DB iff:
   - Response is valid
   - Response is signed

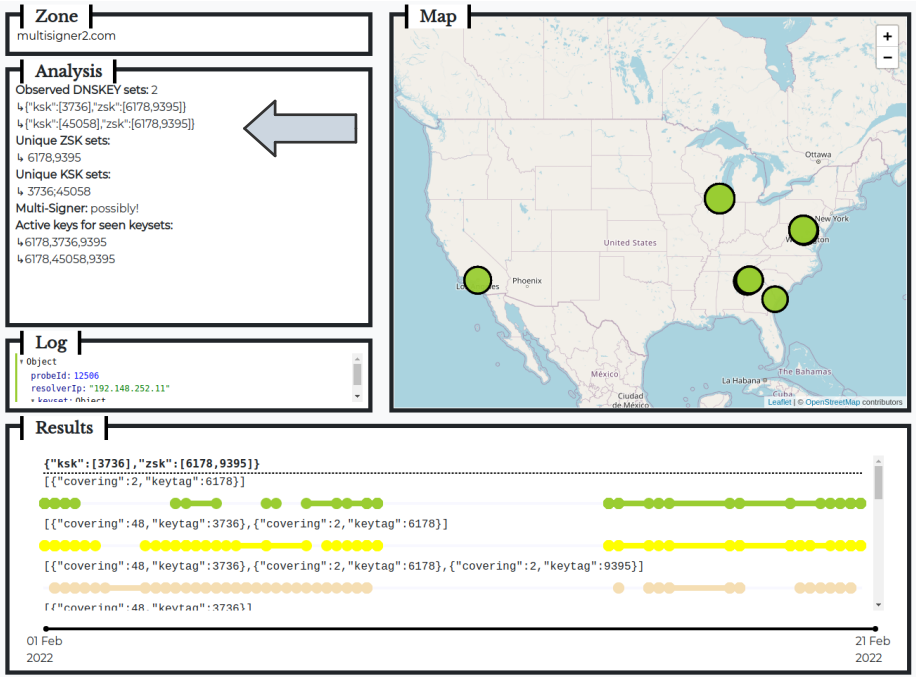
Executed by a set of random probes (currently only US)

Also record when each probes sees which RRSet and RRSIG

Approach: Provide Analysis

For any given zone:

1. Calculate different combinations of *observed* `DNSKEY` sets and active keys in use.



Approach: Provide Analysis

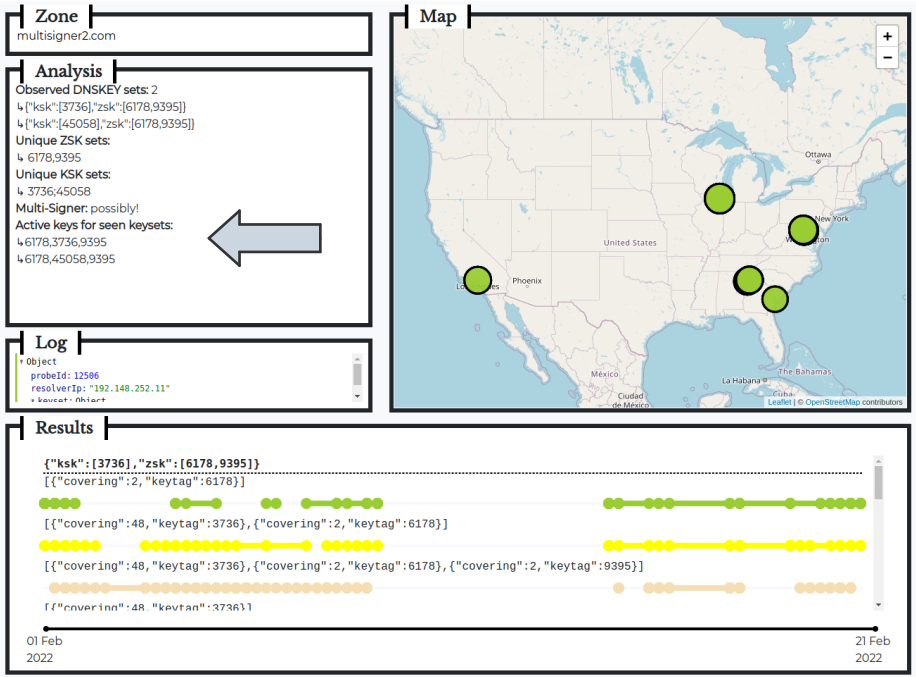
For any given zone:

1. Calculate different combinations of *observed* `DNSKEY` sets and active keys in use.
2. Color code each combination and calculate when each probe sees which combination.

Approach: Provide Analysis

For any given zone:

1. Calculate different combinations of *observed* `DNSKEY` sets and active keys in use.
2. Color code each combination and calculate when each probe sees which combination.
3. Analyze for specific events or deployment models: ongoing key transitions, multi signer DNSSEC, etc.



Conclusion

- There is a measurable discrepancy between records at authoritative name servers and what recursive resolvers deliver
- *DNS(SEC) Views* gives operators the opportunity to follow their DNSSEC deployment from the perspective of clients in real time
- Aggregated data can be used to improve deployment practices and figure out acceptance criteria