Measuring KSK Roll Readiness

Getting resolvers to report on their local trusted key state

• Resolvers that support the RFC8145 signal mechanism periodically include the key tag of their locally trusted keys into a query directed towards the root servers

But:

• An aggregated signal is only visible to root servers
• DNS forwarders and local caching confuse attribution efforts
• The number of users that exclusively rely on reporting resolvers is not apparent
• It is unknown whether the user has alternate resolvers that they can use
User-Side Measurement

Can we devise a DNS query that could reveal the state of the trusted keys of the resolvers that the user actually invokes back to the user?

• Not within the current parameters of DNSSEC and/or resolver behaviour

• But what if we could change resolver behaviour?
  • Just as RFC8145 required a change in resolver behaviour
  • We propose a change to the resolver’s reporting of validation outcome depending on the resolver’s local trusted key state:
    • If a query contains the label “_is-ta-<key-tag>” then a validating resolver will report validation failure if the key is NOT in the local trusted key store
    • If a query contains the label “_not-ta-<key-tag>” then a validating resolver will report validation failure if the key IS in the local trusted key store
User-Side Measurement

Three DNS queries:

1. _is-ta-4066.<some.signed.domain>
2. _not-ta-4066.<some.signed.domain>
3. <badly-signed>.<some.signed.domain>

Single Resolver Analysis:

<table>
<thead>
<tr>
<th>Resolver Behaviour Type</th>
<th>Query 1</th>
<th>Query 2</th>
<th>Query 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loaded New KSK</td>
<td>A</td>
<td>SERVFAIL</td>
<td>SERVFAIL</td>
</tr>
<tr>
<td>NOT loaded New KSK</td>
<td>SERVFAIL</td>
<td>A</td>
<td>SERVFAIL</td>
</tr>
<tr>
<td>Mechanism not supported</td>
<td>A</td>
<td>A</td>
<td>SERVFAIL</td>
</tr>
<tr>
<td>Not validating</td>
<td>A</td>
<td>A</td>
<td>A</td>
</tr>
</tbody>
</table>
User-Side Measurement

Multiple Resolver Analysis

A SERVFAIL response will cause the user to repeat their query to other locally configured resolvers. In a multi-resolver scenario, and where forwarders are used, we can still determine if the user will be impacted by the KSK roll.

<table>
<thead>
<tr>
<th>User Impact</th>
<th>Query 1</th>
<th>Query 2</th>
<th>Query 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>A</td>
<td>SERVFAIL</td>
<td>SERVFAIL</td>
</tr>
<tr>
<td>NOT OK</td>
<td>SERVFAIL</td>
<td>A</td>
<td>SERVFAIL</td>
</tr>
<tr>
<td>UNKNOWN</td>
<td>A</td>
<td>SERVFAIL</td>
<td>SERVFAIL</td>
</tr>
</tbody>
</table>

NOT Impacted   | A       | A       | A
Measuring User Impact

Use these tests in a script to allow users to test the state of their DNS environment:

- If the user can resolve Query 1, and SERVFAILs on Query 2 and Query 3 then the user is **able** to validate using the nominated key as a trusted key.
- If the user SERVFAILS on Query 1, resolves Query 2 and SERVFAILs on Query 3 then the user is **unable** to validate using the nominated key as a trusted keys.
- If the user SERVFAILS on Query 3 then the result is indeterminate.
- Otherwise, the user will not be impacted by the KSK roll.
Privacy and Security Considerations

• This test itself does not reveal which resolvers are used by end users in resolving names
• The query itself need not contain any end user identifying material
• The methodology never changes “insecure” to ”authenticated” – it will only change “authenticated” to “insecure” depending on the resolver’s local trusted key state when resolving certain labels
• Anyone can set up a test condition within their delegated part of the DNS
• The results of the test are passed back only to the user in the form of a resolution outcome
Questions

• Should this label be at any location in the name or should it be specified to be the left-most label?
• I can’t think of any other questions – maybe you can!