DNS $ORIGIN dot

Good afternoon, all!

I'm going over some BIND zone files and have run across an oddity that I haven't found a good answer. Several of these zone files have an ORIGIN directive of just a dot (.). Looks weird, I know. Here's the sanitized directives:

```
$ORIGIN .
$TTL 600    ; 10 minutes
example.com    IN SOA  ns1.example.com. support.example.com. (
               2016010101 ; serial
               28800      ; refresh (8 hours)
               120        ; retry (2 minutes)
               1209600    ; expire (2 weeks)
               86400      ; minimum (1 day)
               )
```

The file seems to work OK and it passes online tests from DNSStuff. I do plan on

Any thoughts from the gurus?

Thanks to all for looking!

G

domain-name-system   bind

Share Improve this question Follow

asked Oct 23, 2017 at 21:13

Gregg Hughes
67 1   4

---

1 Answer

Sorted by: Highest score (default) ▲▼

▲

7

▼

http://www.zytrax.com/books/dns/ch8/origin.html

`$ORIGIN` defines a base name from which 'unqualified' names (those without a terminating dot) substitutions are made when processing the zone file.

When set to only a dot `.` that means that only a dot is going to be added/substituted. That's all.

Typically you would see:

```
$ORIGIN example.com.
@  IN NS   ns1.example.com
@  IN NS   ns2.example.com.
@  IN MX  5 mail
```

Which because the `@` symbol is a shorthand for the $ORIGIN in zone files will be completed into:

```
example.com.  IN  NS   ns1.example.com.example.com.
example.com.  IN  NS   ns2.example.com.
example.com.  IN  MX 5 mail.example.com.
```

Where the first line shows a typical erroneous record that will result of such substitution when the $ORIGIN gets appended to what was intended to be a FQDN which lacked the trailing `.` .

Setting $ORIGIN to a dot makes the use of the @ shorthand impractical but will prevent mishaps such as the top one.

```
$ORIGIN .
example.com   IN  NS  ns1.example.com.
example.com.  IN  NS  ns2.example.com
```

Will autocomplete/correct that into

```
example.com.  IN  NS  ns1.example.com.
example.com.  IN  NS  ns2.example.com.
```

And not into

```
example.com.example.com.  IN  NS  ns1.example.com.
example.com.              IN  NS  ns2.example.com.example.com.
```

which is what would have happened if the $ORIGIN would still have been the `example.com.` domain.