# DN2. DNS: longevidade de nomes

Rafilx

2022-05-02

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## Loading required package: viridisLite

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

## R Markdown

A longevidade de um nome (QNAME+QTYPE) no dataset pode ser definida como o intervalo entre a primeira e a última aparição desse nome. Calcular a longevidade dos nomes no dataset, e analisar como essa variável está distribuída.

Resultados esperados:

- análise gráfica da distribuição (histograma, ECDF) e numérica (min, max, média, mediana) da longevidade dos nomes
  - por enquanto não vejo sentido em dividir a análise por período, então pode considerar o dataset como um todo
  - minha intuição é que a distribuição seja assimétrica com (longa) cauda à direita
- Busca os dados no banco com o parse do DNS ja realizado, então temos:
  - qname que é o domínio
  - QTYPE tipo da query
  - query_id ID da transação definido pelo atacante
  - year_period ano e trimestre em que ocorreu o ataque exemplo "20212" o ataque ocorreu no segundo trimestre do 2021

```
db <- dbConnect(RSQLite::SQLite(), dbname="../dnstor_statistics_dns.sqlite")

data_unfetch <-dbSendQuery(db, "
  SELECT *
    FROM DNS_ANALYSIS
        JOIN DNS_ANALYSIS_QUESTION
          ON DNS_ANALYSIS.id = DNS_ANALYSIS_QUESTION.dns_analysis_id
     WHERE QTYPE != 0
")
data <- fetch(data_unfetch)

dbDisconnect(db)
```

```
## Warning in connection_release(conn@ptr): There are 1 result in use. The
## connection will be released when they are closed
```

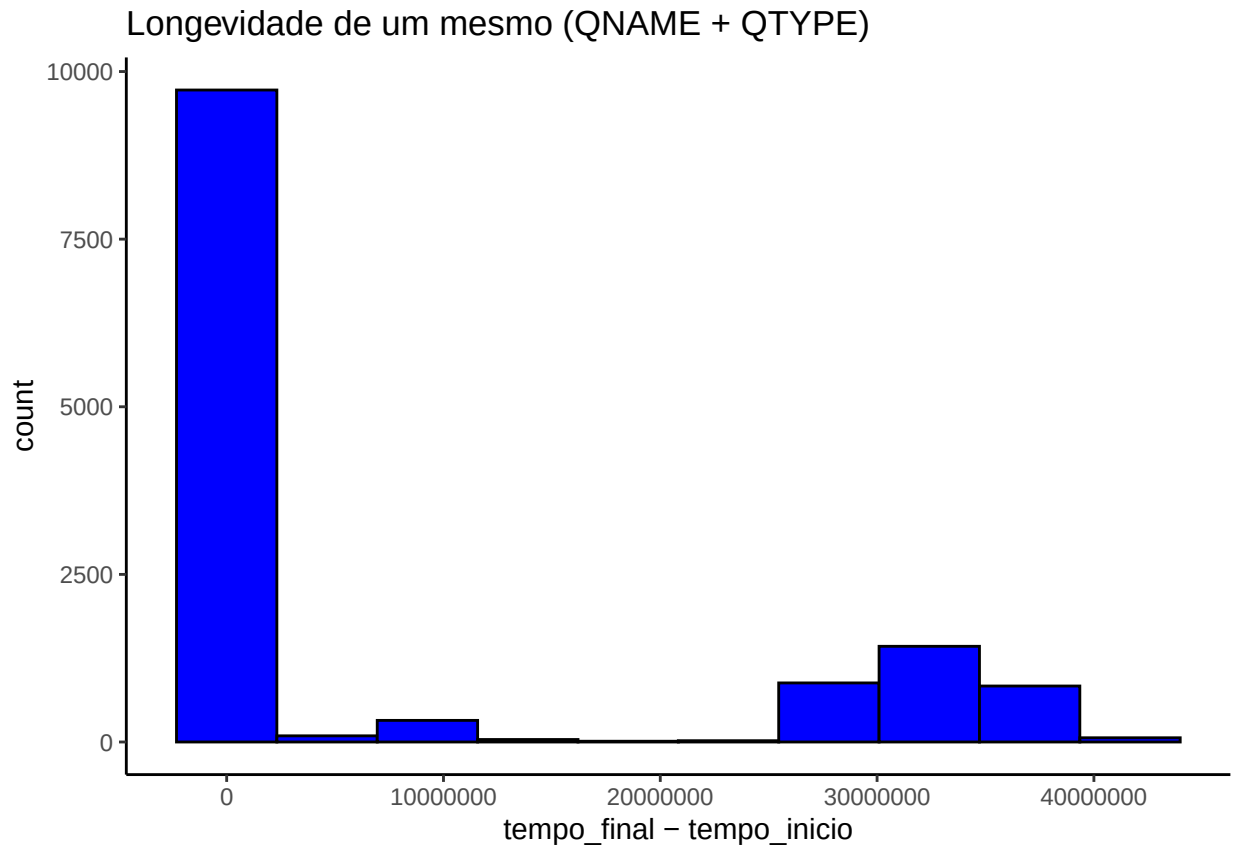
```
data['tempo_final_cast'] = as.POSIXct(data[['tempo_final']], format = "%Y-%m-%d %H:%M:%S")
data['tempo_inicio_cast'] = as.POSIXct(data[['tempo_inicio']], format = "%Y-%m-%d %H:%M:%S")

secs_to_month = (60 * 60 * 24 * 30)

data_grouped = data %>%
  group_by(qname, qtype) %>%
  summarise(tempo_inicio=min(tempo_inicio_cast), tempo_final=max(tempo_final_cast)) %>%
  mutate(tempo_diff_secs = as.numeric(tempo_final - tempo_inicio, units="secs"), tempo_diff = tempo_fina
  #filter(tempo_diff > 0) %>%
  arrange(desc(tempo_diff_secs))
```

```
## `summarise()` has grouped output by 'qname'. You can override using the
## `.groups` argument.
```

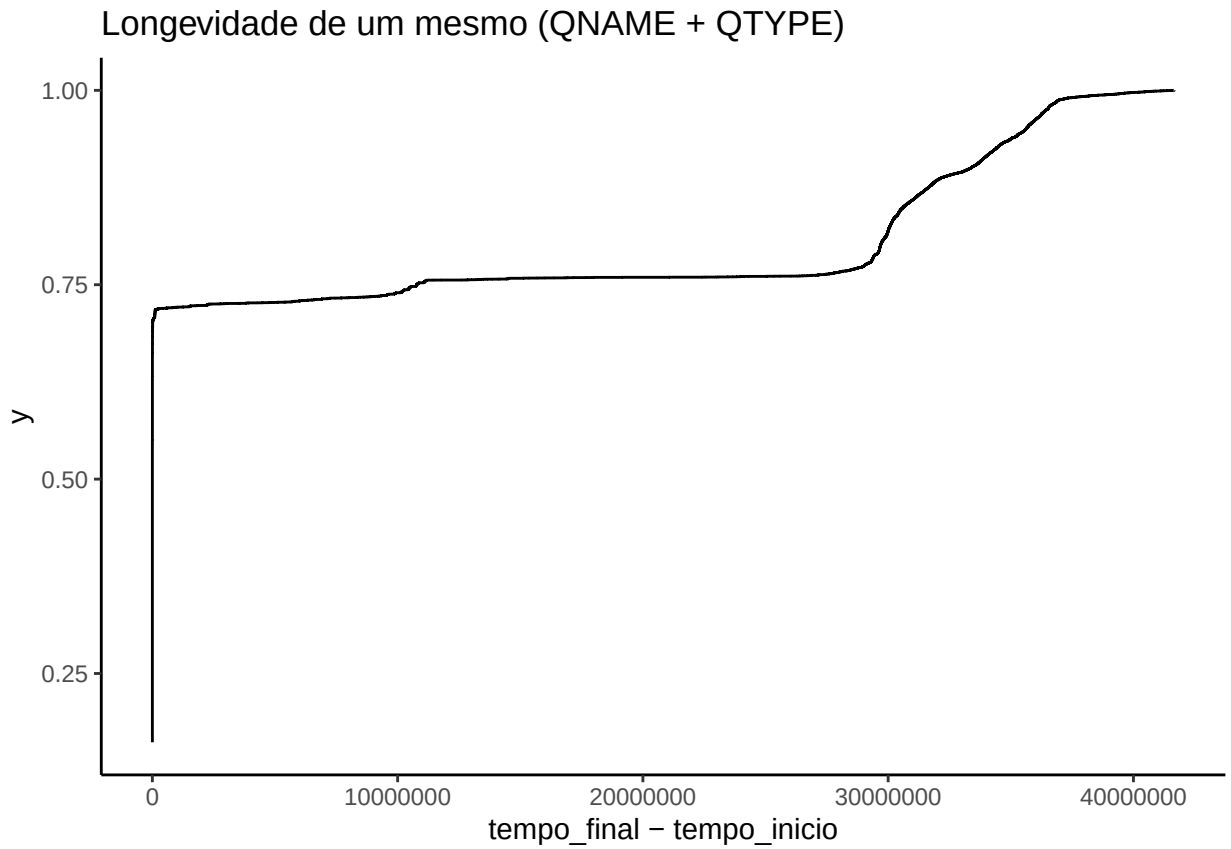
```
data_grouped %>%
  head(10)
```

```
## # A tibble: 10 x 6
## # Groups:   qname [10]
##    qname           qtype tempo_inicio        tempo_final         tempo_diff_secs
##    <chr>           <chr> <dttm>              <dttm>                        <dbl>
##  1 VERSION.BIND.   TXT   2020-10-30 02:39:27 2022-02-24 08:33:01        41666014
##  2 whoami.akamai.~ A     2020-10-30 12:33:16 2022-02-23 10:35:19        41551323
##  3 com.            ANY   2020-10-31 11:38:32 2022-02-24 09:16:50        41549898
##  4 version.bind.   TXT   2020-11-01 04:46:10 2022-02-24 15:42:17        41511367
##  5 isc.org.        ANY   2020-11-01 22:40:23 2022-02-24 02:25:52        41399129
##  6 researchscan54~ A     2020-11-01 15:23:29 2022-02-23 14:06:13        41380964
##  7 adsports.ae.    MX    2020-11-03 10:59:02 2022-02-23 23:36:50        41258268
##  8 public1.114dns~ A     2020-11-04 09:47:54 2022-02-24 15:42:14        41234060
##  9 238.107.19.200~ PTR   2020-11-01 23:58:38 2022-02-21 13:45:36        41176018
## 10 a.gtld-servers~ A     2020-10-30 02:55:07 2022-02-18 00:20:55        41117148
## # ... with 1 more variable: tempo_diff <drtn>
```

```
data_grouped %>%
  ggplot(aes(x= tempo_diff_secs)) +
  geom_histogram(bins = 10, fill='blue', color ='black') +
  ggtitle("Longevidade de um mesmo (QNAME + QTYPE)") +
  xlab("tempo_final - tempo_inicio") +
  theme_classic()
```




```
data_grouped %>%
  ggplot(aes(x= tempo_diff_secs)) +
  stat_ecdf(geom = "step", pad = FALSE) +
  ggtitle("Longevidade de um mesmo (QNAME + QTYPE)") +
  xlab("tempo_final - tempo_inicio") +
  theme_classic()
```




```
data_grouped.tempo_diff_secs.min = min(data_grouped$tempo_diff_secs)
data_grouped.tempo_diff_secs.max = max(data_grouped$tempo_diff_secs)
data_grouped.tempo_diff_secs.mean = mean(data_grouped$tempo_diff_secs)
data_grouped.tempo_diff_secs.median = median(data_grouped$tempo_diff_secs)

quantile(data_grouped$tempo_diff_secs)
```

```
##       0%      25%      50%      75%     100%
##        0       14       69 10780940 41666014
```

```
summary(data_grouped)
```

```
##     qname              qtype            tempo_inicio
##  Length:13411       Length:13411       Min.   :2020-10-29 16:15:05
##  Class :character   Class :character   1st Qu.:2020-12-13 09:03:49
##  Mode  :character   Mode  :character   Median :2021-01-01 02:01:27
##                                        Mean   :2021-04-27 18:46:19
##                                        3rd Qu.:2021-10-25 18:19:09
##                                        Max.   :2022-02-24 15:42:16
##   tempo_final                  tempo_diff_secs     tempo_diff
##  Min.   :2020-10-29 23:17:13   Min.   :       0   Length:13411
##  1st Qu.:2020-12-15 00:54:54   1st Qu.:      14   Class :difftime
##  Median :2021-10-25 06:53:39   Median :      69   Mode  :numeric
##  Mean   :2021-07-31 07:23:43   Mean   : 8167044
```

```
##  3rd Qu.:2021-12-06 22:50:36   3rd Qu.:10780940
##  Max.   :2022-02-24 15:49:24   Max.   :41666014
```

- Dados sobre o intervalo entre a primeira e a última aparição desse (QNAME+QTYPE)
  - Mínimo 0 segundos
  - Máximo 16.0749 meses
  - Média 136117.3974 minutos
  - Mediana 69 segundos

```
trim_value = .30

data_grouped.tempo_diff_secs.min = min(data_grouped$tempo_diff_secs, trim=trim_value)
data_grouped.tempo_diff_secs.max = max(data_grouped$tempo_diff_secs, trim=trim_value)
data_grouped.tempo_diff_secs.mean = mean(data_grouped$tempo_diff_secs, trim=trim_value)
data_grouped.tempo_diff_secs.median = median(data_grouped$tempo_diff_secs, trim=trim_value)

quantile(data_grouped$tempo_diff_secs, trim=trim_value)
```

```
##       0%      25%      50%      75%     100%
##        0       14       69 10780940 41666014
```

```
summary(data_grouped, trim=trim_value)
```

```
##     qname              qtype            tempo_inicio
##  Length:13411       Length:13411       Min.   :2020-10-29 16:15:05
##  Class :character   Class :character   1st Qu.:2020-12-13 09:03:49
##  Mode  :character   Mode  :character   Median :2021-01-01 02:01:27
##                                        Mean   :2021-04-27 18:46:19
##                                        3rd Qu.:2021-10-25 18:19:09
##                                        Max.   :2022-02-24 15:42:16
##   tempo_final                  tempo_diff_secs     tempo_diff
##  Min.   :2020-10-29 23:17:13   Min.   :       0   Length:13411
##  1st Qu.:2020-12-15 00:54:54   1st Qu.:      14   Class :difftime
##  Median :2021-10-25 06:53:39   Median :      69   Mode  :numeric
##  Mean   :2021-07-31 07:23:43   Mean   : 8167044
##  3rd Qu.:2021-12-06 22:50:36   3rd Qu.:10780940
##  Max.   :2022-02-24 15:49:24   Max.   :41666014
```

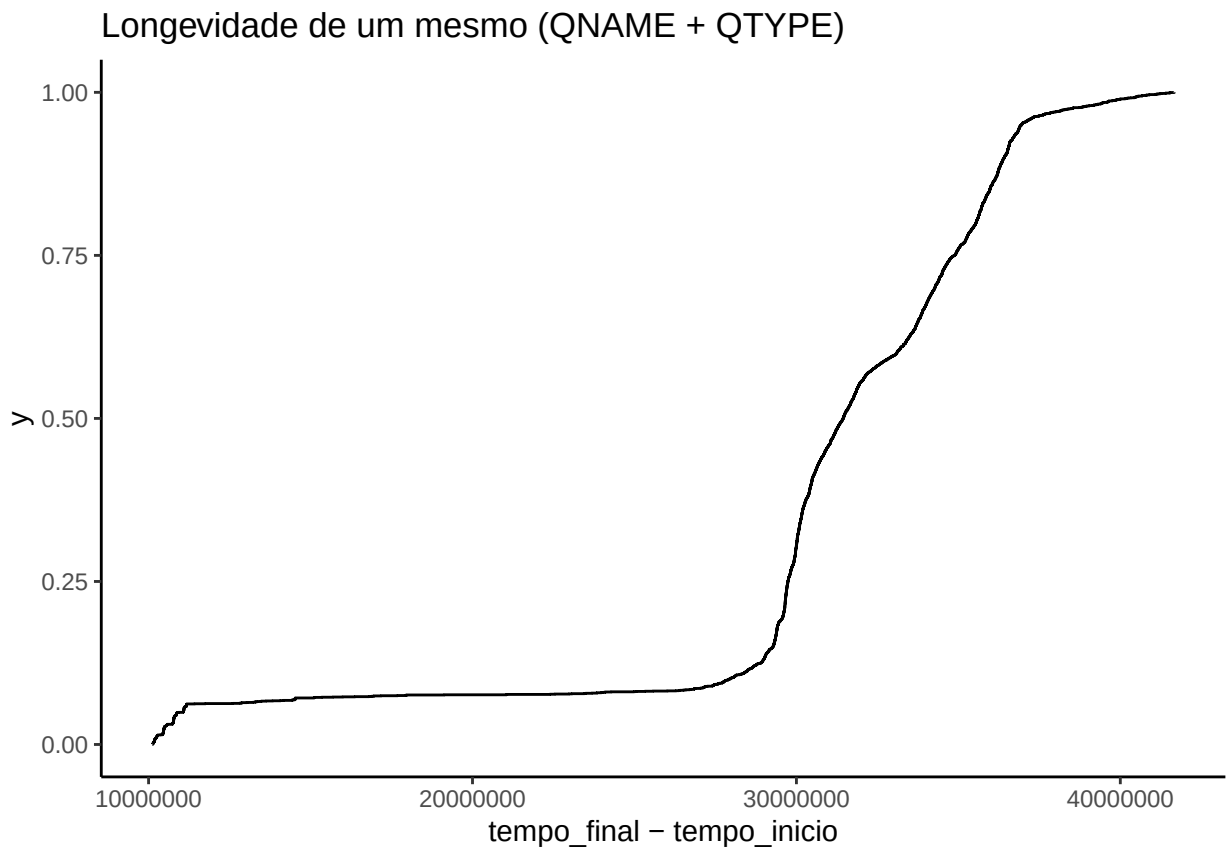
- Dados sobre o intervalo entre a primeira e a última aparição do (QNAME+QTYPE) removendo 30% dos valores máximos e mínimos
  - Mínimo 0 segundos
  - Máximo 16.0749 meses
  - Média 12.7382 minutos
  - Mediana 69 segundos
- Identificar qual é o percentil em que há essa mudança de tendência (próximo aos 72-73%) e qual a duração correspondente

```
quantile(data_grouped$tempo_diff_secs,
         c(.5332, .6235, .696282, .710307, .72, .7204,
           .721090026, .73, .7396902445, .75, .76,
           .77, .78, .99))
```

```
##   53.32%   62.35%   69.63%   71.03%      72%   72.04%   72.11%      73%
##      100     1000    10000   100000   598289   803076  1000000  6310057
##   73.97%      75%      76%      77%      78%      99%
## 10000000 10780940 23306177 28575350 29310516 37308538
```

- Isso representa que Y% dos (QNAME+TYPE), tem os seus ataques com duração de até X segundos:
  - 53.3% até 100 segundos
  - 62.3% até 1000 segundos
  - 69.6% até 10000 segundos
  - 71% até 100000 segundos
  - 72% até 598289 segundos
  - 72.04% até 803076 segundos
  - 72.11% até 1000000 segundos
  - 73.97% até 10000000 segundos
- Significa que após os 69% o tempo dos ataques cresce muito até cerca de 73.97% onde estabiliza próximo dos 10000000 segundos
- Uma representação ECDF removendo os registros abaixo da quantidade de segundos em que apresenta estabilidade (10000000 segundos)
  - Possívelmente apresenta uma distribuição assimétrica com cauda a direita

```
data_grouped %>%
  filter(tempo_diff_secs > 10000000) %>%
  ggplot(aes(x= tempo_diff_secs)) +
  stat_ecdf(geom = "step", pad = FALSE) +
  ggtitle("Longevidade de um mesmo (QNAME + QTYPE)") +
  xlab("tempo_final - tempo_inicio") +
  theme_classic()
```




```
percentage_76_secs = quantile(data_grouped$tempo_diff_secs, c(.76))[[1]]
percentage_76_secs/secs_to_month
```

```
## [1] 8.992
```

```
percentage_99_secs = quantile(data_grouped$tempo_diff_secs, c(.995))[[1]]
percentage_99_secs/secs_to_month
```

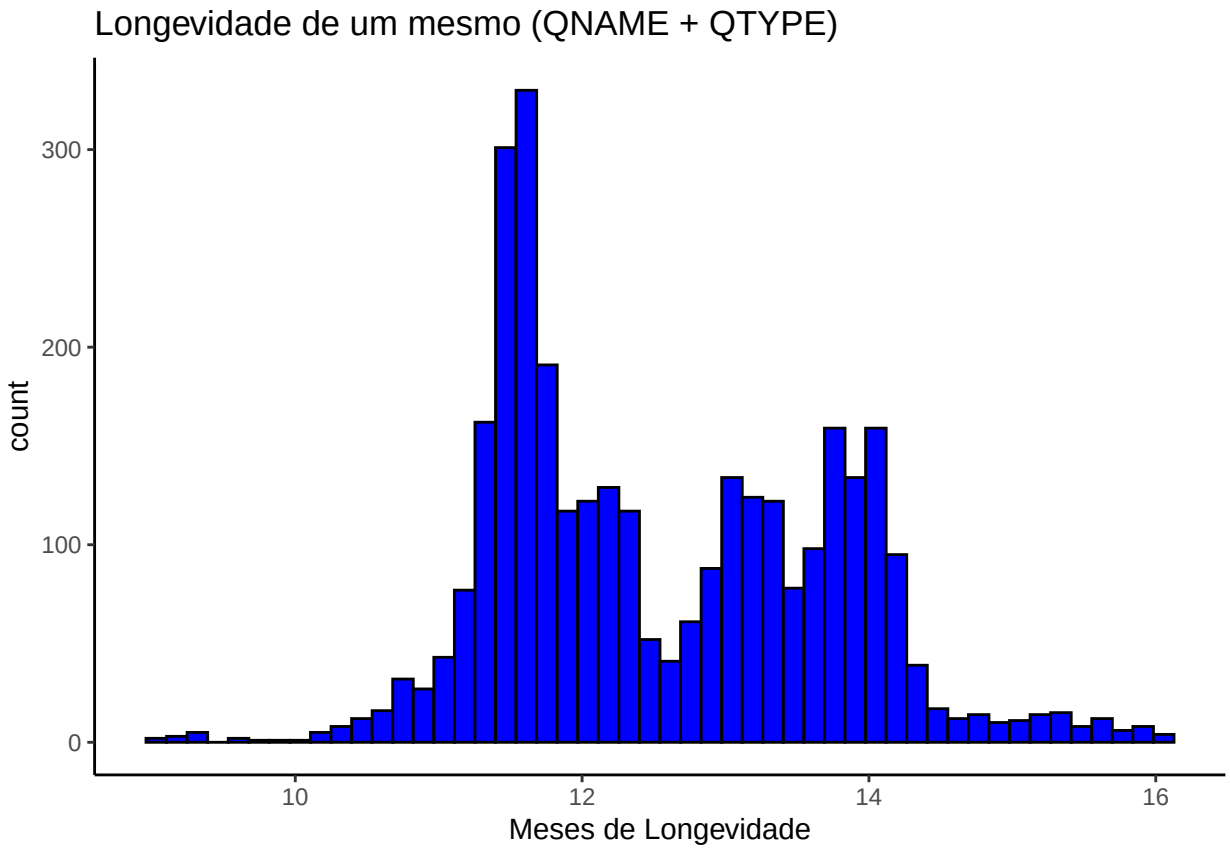
```
## [1] 15.12
```

```
data_grouped.tempo_diff_secs.max / secs_to_month
```

```
## [1] 16.07
```

- Cerca de 24% dos (QNAME + QTYPE) possuem uma longevidade entre 9 e 16 meses

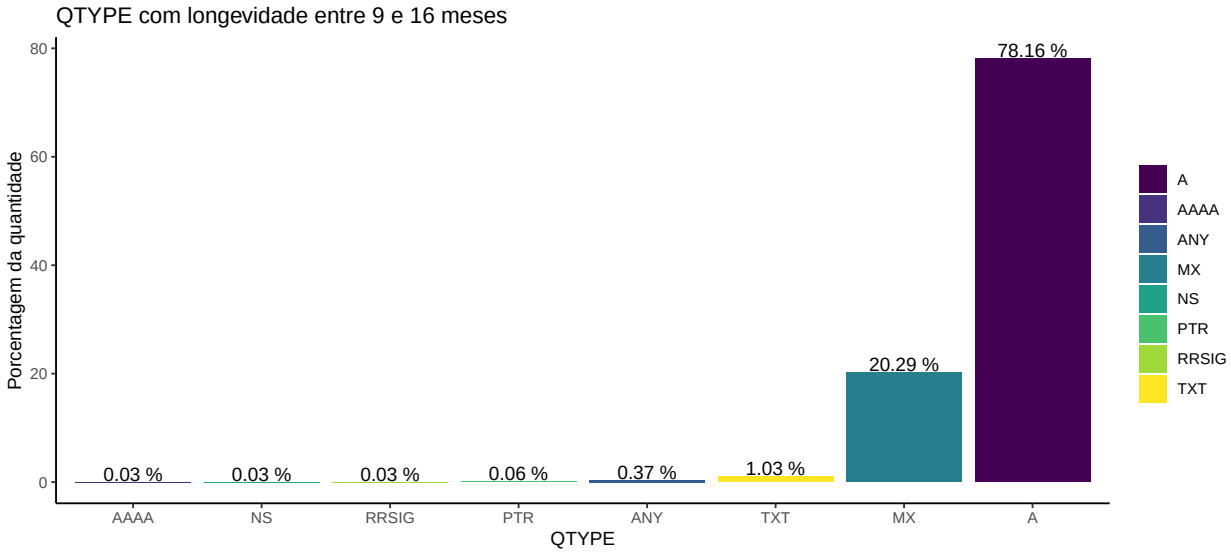
```
data_grouped %>%
  filter(tempo_diff_secs > percentage_76_secs) %>%
  ggplot(aes(x= tempo_diff_secs / secs_to_month)) +
  geom_histogram(bins = 50, fill='blue', color ='black') +
  ggtitle("Longevidade de um mesmo (QNAME + QTYPE)") +
  xlab("Meses de Longevidade") +
  theme_classic()
```




```
data_bigger_than_76 = data_grouped %>%
  filter(tempo_diff_secs > percentage_76_secs) %>%
  ungroup() %>%
  group_by(qtype) %>%
  summarise(qtype_quantity = n()) %>%
  arrange(desc(qtype_quantity))


sum_qtype_quantity = sum(data_bigger_than_76$qtype_quantity)
data_bigger_than_76_percentage = data_bigger_than_76 %>%
  mutate(qtype_quantity_percentage = (qtype_quantity / sum_qtype_quantity) * 100)

data_bigger_than_76_percentage
```

```
## # A tibble: 8 x 3
##   qtype qtype_quantity qtype_quantity_percentage
##   <chr>          <int>                     <dbl>
## 1 A               2516                    78.2
## 2 MX               653                    20.3
## 3 TXT               33                     1.03
## 4 ANY               12                     0.373
## 5 PTR                2                     0.0621
## 6 AAAA               1                     0.0311
## 7 NS                 1                     0.0311
## 8 RRSIG              1                     0.0311
```

```r
data_bigger_than_76_percentage %>%
 ggplot( aes(x=reorder(qtype, +qtype_quantity_percentage), y=qtype_quantity_percentage, fill=qtype)) +
    geom_bar(stat="identity", position="dodge") +
    scale_fill_viridis(discrete=TRUE, name="") +
    geom_text(aes(label = paste(round(qtype_quantity_percentage, 2), "%")), vjust = -0.10, ) +
    theme_classic() +
    ylab("Porcentagem da quantidade") +
    xlab("QTYPE") +
    ggtitle("QTYPE com longevidade entre 9 e 16 meses")
```




- Então dos QTYPE que possuem uma alta longevidade entre 9 e 16 meses (cerca de 24% de todos os registros 76% ~ 100%)
  - 20% (653) deles possuem o QTYPE "MX"
  - 78% (2516) dos ataques com maior longevidade utilizam o QTYPE "A", o que é surpreendente
  - E por fim o QTYPE "ANY" aparece com apenas 12 registros de QTYPE com longevidade entre 9 e 16 meses