Stay safe out there, folks; and — to my not-so-'United'-after-all States readers — stay strong! The nightmare of the last four years is almost over (though the cleanup — now both physical and metaphorical — is going to take a long time).

```
library(urltools)
library(stringi)
library(tidyverse)
# we're also using {clipr} and {tools} but via ::: and ::

# fairly comprehensive list of URL shorteners
shorteners <- read_lines("https://github.com/sambokai/ShortURL-Services-List/raw
/master/shorturl-services-list.txt")

# opaque function baked into {tools}
# NOTE: this can take a while
db <- tools:::url_db_from_installed_packages(rownames(installed.
packages()), verbose = TRUE)

as_tibble(db) %>%
  distinct() %>%  # yep, even w/in a pkg there may be dups from ^^
  mutate(
    scheme = scheme(URL), # https or not
    dom = domain(URL)     # need this later to be able to compute apex
domain
  )  %>%
  filter(
    dom != "..", # prbly legit since it will be a relative "go up one
directory"
    !is.na(dom)  # the {tools} url_db_from_installed_packages() is not
perfect
  ) %>%
  bind_cols(
    suffix_extract(.$dom) # break them all down into component atoms
  ) %>%
  select(-dom) %>% # this is now 'host' from ^^
  mutate(
    apex = sprintf("%s.%s", domain, suffix) # apex domain
  ) %>%
  mutate(
    is_short = (host %in% shorteners) | (apex %in% shorteners) # does
it use a shortener?
  ) -> db

db
## # A tibble: 12,623 x 9
##    URL        Parent    scheme host  subdomain domain suffix apex
is_short
##
##  1 https://g… albersus… https  gith… NA        github com    gith…
FALSE
```

```
##  2 https://g… albersus… https  gith… NA        github com    gith…
FALSE
##  3 https://w… AnomalyD… https  www.… www       usenix org    usen…
FALSE
##  4 https://w… AnomalyD… https  www.… www       jstor  org    jsto…
FALSE
##  5 https://w… AnomalyD… https  www.… www       usenix org    usen…
FALSE
##  6 https://w… AnomalyD… https  www.… www       jstor  org    jsto…
FALSE
##  7 https://g… AnomalyD… https  gith… NA        github com    gith…
FALSE
##  8 https://g… AnomalyD… https  gith… NA        github com    gith…
FALSE
##  9 https://g… AnomalyD… https  gith… NA        github com    gith…
FALSE
## 10 https://g… AnomalyD… https  gith… NA        github com    gith…
FALSE
## # … with 12,613 more rows

# what packages do i have installed that use short URLS?
# a nice thing to do would be to file a PR to these authors

filter(db, is_short) %>%
  select(
    URL,
    Parent,
    scheme
  )
## # A tibble: 5 x 3
##   URL                          Parent                    scheme
##
## 1 https://goo.gl/5KBjL5        fpp2/man/goog.Rd          https
## 2 http://bit.ly/2016votecount  geofacet/man/election.Rd http
## 3 http://bit.ly/SnLi6h         knitr/man/knit.Rd         http
## 4 https://bit.ly/magickintro   magick/man/magick.Rd      https
## 5 http://bit.ly/2UaiYbo        ssh/doc/intro.html        http

# what protocols are in use? (you'll note that some are borked and
# others got mangled by the {tools} function)

count(db, scheme, sort=TRUE)
## # A tibble: 5 x 2
##   scheme     n
##
## 1 https  10007
## 2 http    2498
## 3 NA       113
## 4 ftp        4
## 5 `https     1
```

```
# what are the most used top-level sites?

count(db, host, sort=TRUE) %>%
  mutate(pct = n/sum(n))
## # A tibble: 1,108 x 3
##    host                       n     pct
##
##  1 docs.aws.amazon.com     3859 0.306
##  2 github.com              2954 0.234
##  3 cran.r-project.org       450 0.0356
##  4 en.wikipedia.org         220 0.0174
##  5 aws.amazon.com           204 0.0162
##  6 doi.org                  181 0.0143
##  7 wikipedia.org            132 0.0105
##  8 developers.google.com    114 0.00903
##  9 stackoverflow.com        101 0.00800
## 10 gitlab.com                86 0.00681
## # … with 1,098 more rows

# same as ^^ but apex

count(db, apex, sort=TRUE) %>%
  mutate(pct = n/sum(n))
## # A tibble: 743 x 3
##    apex                   n     pct
##
##  1 amazon.com          4180 0.331
##  2 github.com          2997 0.237
##  3 r-project.org        563 0.0446
##  4 wikipedia.org        352 0.0279
##  5 doi.org              221 0.0175
##  6 google.com           179 0.0142
##  7 tidyverse.org        151 0.0120
##  8 r-lib.org            137 0.0109
##  9 rstudio.com          117 0.00927
## 10 stackoverflow.com    102 0.00808
## # … with 733 more rows

# See all the eavesdroppable, interceptable,
# content-mutable-by-evil-MITM-network-operator URLs
# A nice thing to do would be to fix these and issue PRs

filter(db, scheme == "http") %>%
  select(URL, Parent)
## # A tibble: 2,498 x 2
##    URL                                                Parent
##
##  1 http://www.winfield.demon.nl                       antiword/DESCRIPTION
##  2 http://github.com/ropensci/antiword/issues         antiword/DESCRIPTION
##  3 http://dirk.eddelbuettel.com/code/anytime.html     anytime/DESCRIPTION
```

```
##  4 http://arrayhelpers.r-forge.r-project.org/          arrayhelpers/DESCRI…
##  5 http://arrow.apache.org/blog/2019/01/25/r-spark-im… arrow/doc/arrow.html
##  6 http://docs.aws.amazon.com/AmazonS3/latest/API/RES… aws.s3/man/accelera…
##  7 http://docs.aws.amazon.com/AmazonS3/latest/API/RES… aws.s3/man/accelera…
##  8 http://docs.aws.amazon.com/AmazonS3/latest/dev/acl… aws.s3/man/acl.Rd
##  9 http://docs.aws.amazon.com/AmazonS3/latest/API/RES… aws.s3/man/bucket_e…
## 10 http://docs.aws.amazon.com/AmazonS3/latest/API/RES… aws.s3/man/bucketli…
## # … with 2,488 more rows

# find the abusers of "http" URLs

filter(db, scheme == "http") %>%
  select(URL, Parent) %>%
  mutate(
    pkg = stri_match_first_regex(Parent, "(^[^/]+)")[,2]
  ) %>%
  count(pkg, sort=TRUE)
## # A tibble: 265 x 2
##    pkg                        n
##
##  1 paws.security.identity   258
##  2 paws.management          152
##  3 XML                      129
##  4 paws.analytics            78
##  5 stringi                   70
##  6 paws                      57
##  7 RCurl                     51
##  8 igraph                    49
##  9 base                      47
## 10 aws.s3                    44
## # … with 255 more rows

# send all the apex domains to the clipboard

clipr::write_clip(unique(db$apex))

# go here to paste them into the domain search box
# most domain/URL checker APIs aren't free for more
# than a cpl dozen URLs/domains

browseURL("https://www.bulkblacklist.com")

# paste what you clipped into the box and wait a while…
```