part1_markdown

This is a markdown version of the script for part 1 with the plots enables in pdf format the code for the Rscript goes as followed:

```
options( warn = -1 )
library(verisr)
library(ggplot2)
library(dplyr)
```

```
##
## 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
```

```
jsondir <- '~/Desktop/ch07/data/vcdb/'

vcdb_orginal <- json2veris(jsondir)
```

```
## [1] "veris dimensions"
## [1] 1643 1890
##       discovery_method.Ext - unrelated party
##                                         1838
##      discovery_method.Int - reported by user
##                                         1839
##           action.misuse.variety.Embezzlement
##                                         1840
##              discovery_method.Int - IT audit
##                                         1841
## attribute.integrity.variety.Misappropriation
##                                          1842
##     action.physical.vector.Bypassed controls
##                                          1843
##     action.physical.vector.Disabled controls
##                                          1844
## named integer(0)
```

```
summary(vcdb_orginal)
```

```
## 1643 incidents in this object.
```

```
##       actor      action           asset              attribute
##  External:955   3   :398   Kiosk/Term: 17   Availability   : 614
##  Internal:535   4   :216   Media     :520   Confidentiality:1604
##  Partner :100   5   :508   Network   :  8   Integrity      : 165
##  Unknown : 85   6   : 31   Person    : 33
##                 7   :416   Server    :639
##                 8   : 42   Unknown   : 80
##                            User Dev  :407
##
```

```
newjsondir <- '~/Desktop/VCDB-master/data/json/'

vcdb_verizon <- json2veris(newjsondir)
```

```
## [1] "veris dimensions"
## [1] 5711 1879
## named integer(0)
## named integer(0)
```

```
summary(vcdb_verizon)
```

```
## 5711 incidents in this object.
```

```
##       actor                action            asset
##  External:2902   Environmental:    7   Kiosk/Term: 144
##  Internal:2490   Error        :1674   Media     :1662  
##  Partner : 234   Hacking      :1614   Network   : 135  
##  Unknown : 183   Malware      : 428   Person    : 377  
##                  Misuse       :1022   Server    :2490  
##                  Physical     :1104   Unknown   : 524  
##                  Social       : 348   User Dev  :1197  
##                  Unknown      : 203                    
##                                                        
##           attribute   
##  Availability   :1894  
##  Confidentiality:5230  
##  Integrity      :1231  
##                        
##                        
##                        
##                        
##                        
##
```

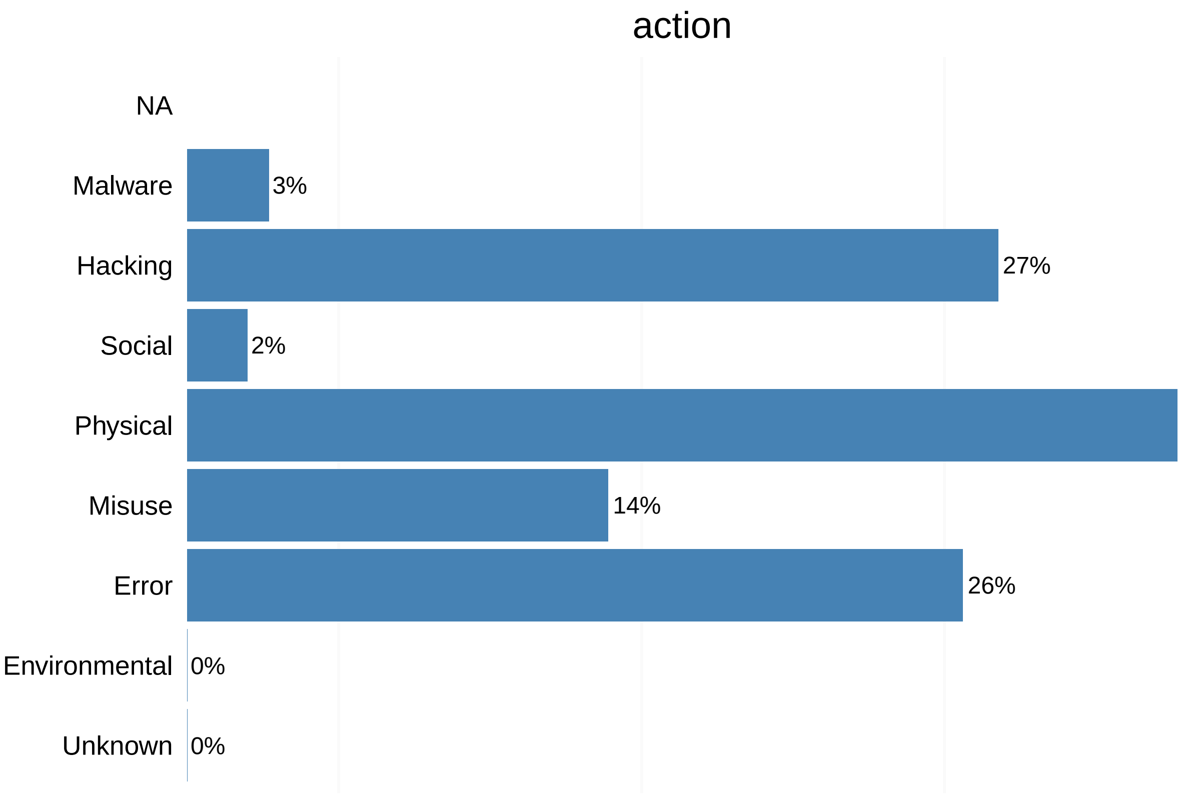
```r
actors <- getenum(vcdb_orginal, "actor")

print(actors)
```

```
##         enum   x    n       freq
## 1: External 955 1640 0.58231707
## 2: Internal 535 1640 0.32621951
## 3:  Partner 100 1640 0.06097561
## 4:  Unknown  85 1640 0.05182927
```

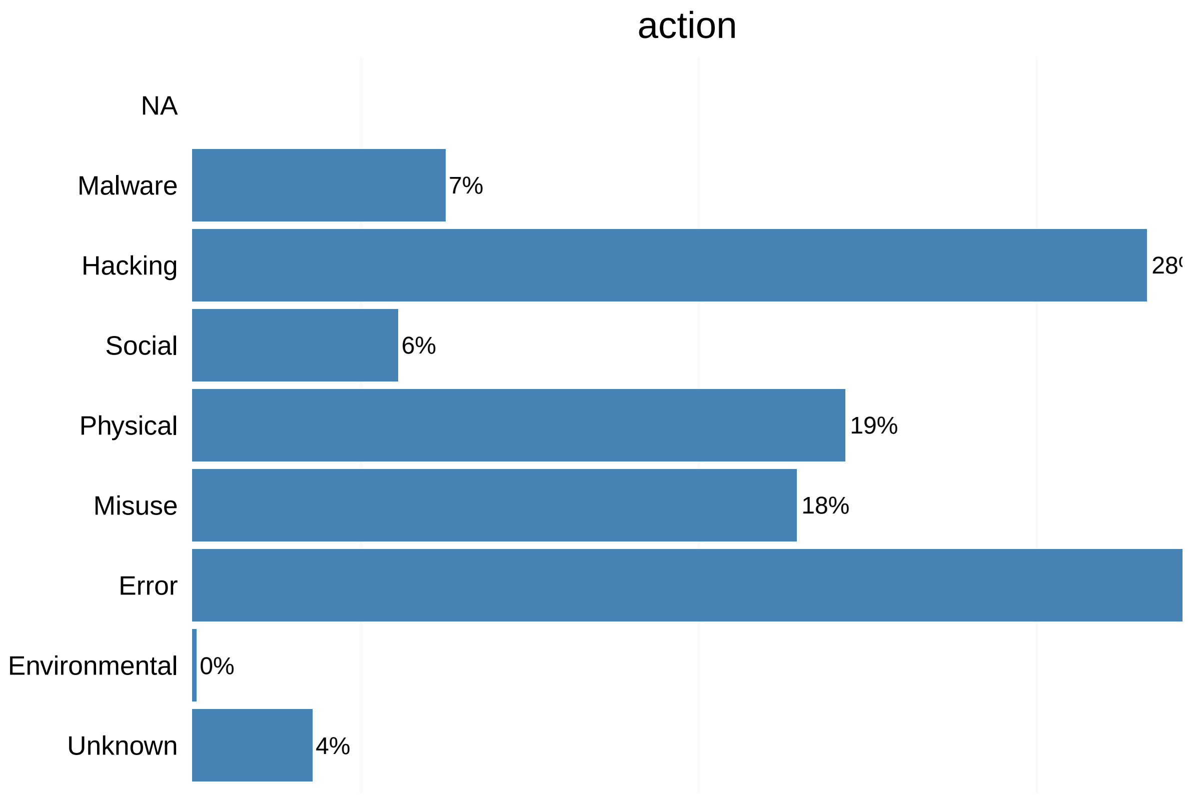
```r
actors2 <- getenum(vcdb_verizon, "actor")

print(actors2)
```

```
##         enum    x    n       freq
## 1: External 2902 5711 0.50814218
## 2: Internal 2490 5711 0.43600070
## 3:  Partner  234 5711 0.04097356
## 4:  Unknown  183 5711 0.03204342
```

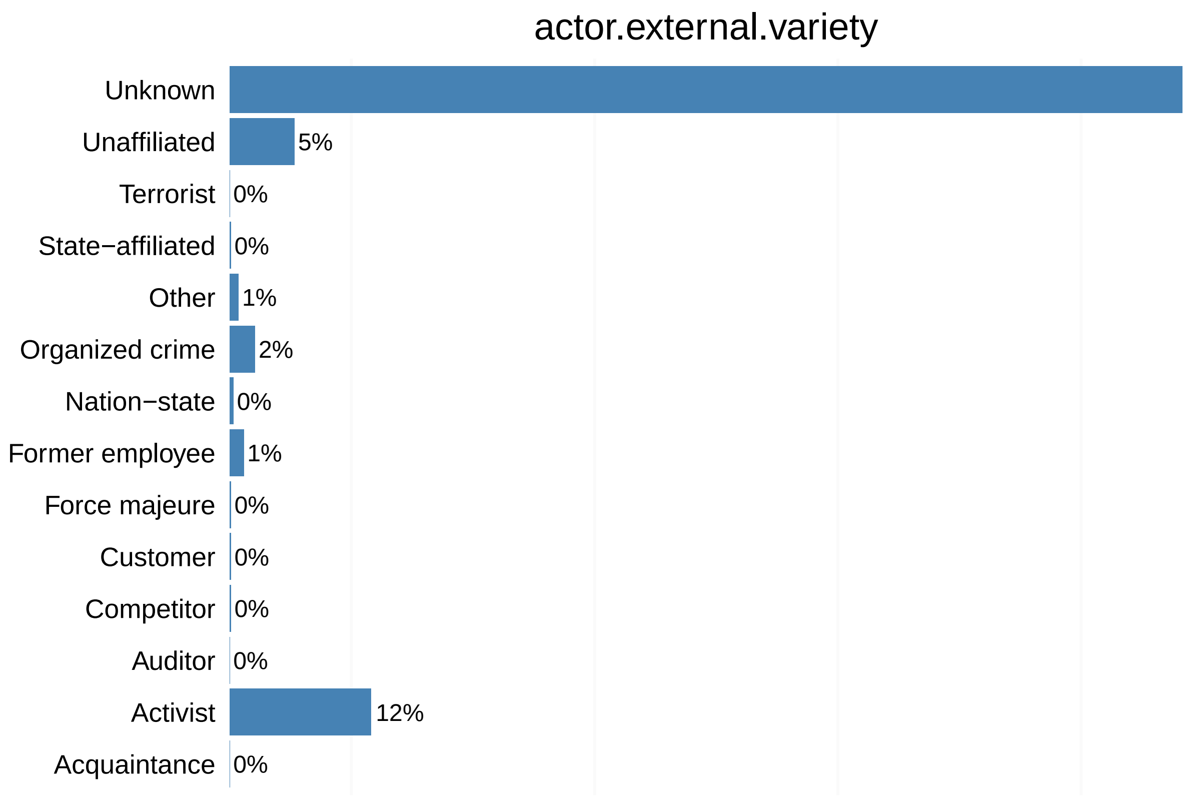
```
verisplot <- function(vcdb_orginal, field) {
  # data frame with field freq
  localdf <- getenum(vcdb_orginal, field, add.freq=T)
  # data set pf first 5
  localdf <- localdf[c(1:15), ]
  # add a label to df
  localdf$lab <- paste(round(localdf$freq*100, 0), "%", sep="")
  # create ggplot
  gg <- ggplot(localdf, aes(x=enum, y=freq, label=lab))
  gg <- gg + geom_bar(stat="identity", fill="steelblue")
  # add in text, adjusted to the end of the bar
  gg <- gg + geom_text(hjust=-0.1, size=3)
  # flip the axes and add in a title
  gg <- gg + coord_flip() + ggtitle(field)
  # remove axes labels and add bw theme
  gg <- gg + xlab("") + ylab("") + theme_bw()
  # fix the y scale to remove padding and fit our label (add 7%)
  gg <- gg + scale_y_continuous(expand=c(0,0),
                                limits=c(0, max(localdf$freq)*1.1))
  # make it slightly prettier than the default
  gg <- gg + theme(panel.grid.major = element_blank(),
                   panel.border = element_blank(),
                   axis.text.x = element_blank(),
                   axis.ticks = element_blank())
}

print (verisplot(vcdb_orginal, "action"))
```



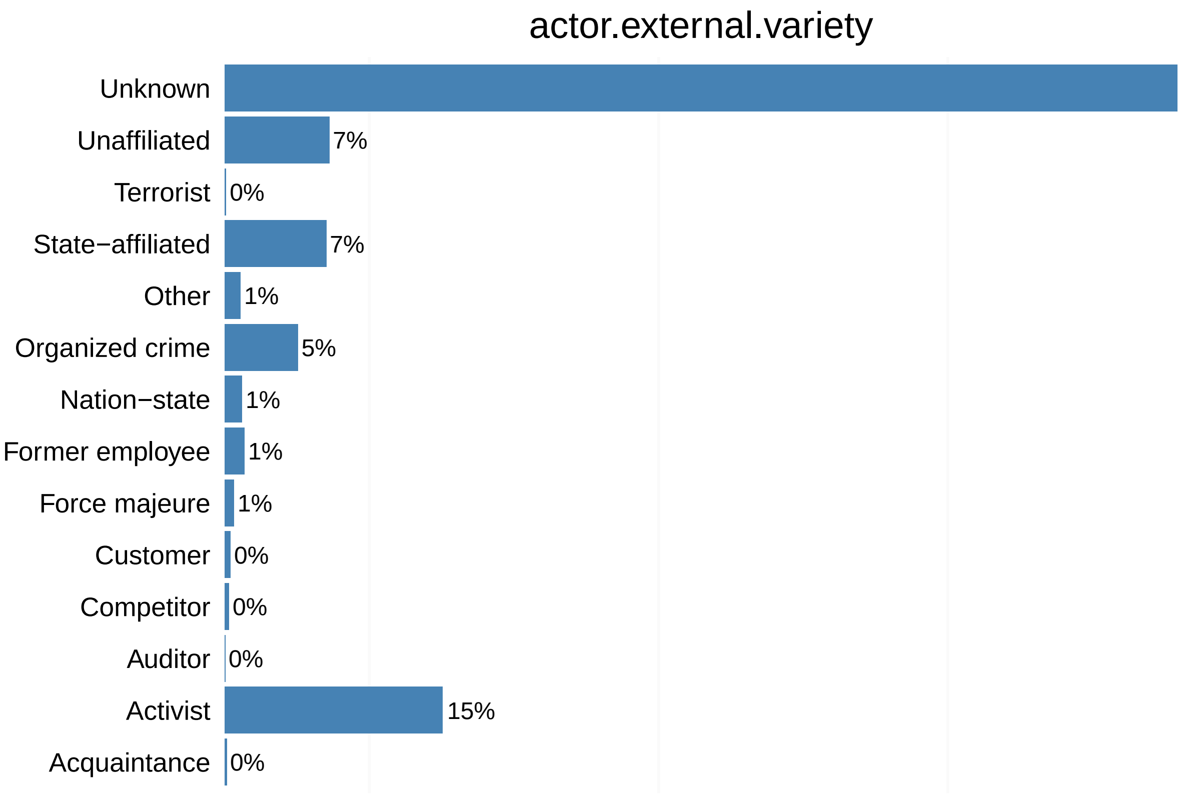
```
verisplot2 <- function(vcdb_verizon, field) {
  # get the data.frame for the field with frequency
  localdf2 <- getenum(vcdb_verizon, field, add.freq=T)
  # now let's take first 5 fields in the data frame.
  localdf2 <- localdf2[c(1:15), ]
  # add a label to the data.frame
  localdf2$lab <- paste(round(localdf2$freq*100, 0), "%", sep="")
  # create a ggplot
  gg2 <- ggplot(localdf2, aes(x=enum, y=freq, label=lab))
  gg2 <- gg2 + geom_bar(stat="identity", fill="steelblue")
  # add text at end of bar
  gg2 <- gg2 + geom_text(hjust=-0.1, size=3)
  #  yx axis and add in a title
  gg2 <- gg2 + coord_flip() + ggtitle(field)
  # remove axis labels
  gg2 <- gg2 + xlab("") + ylab("") + theme_bw()
  # add padding
  gg2 <- gg2 + scale_y_continuous(expand=c(0,0),
                                  limits=c(0, max(localdf2$freq)*1.1))
  # clean up panel
  gg2 <- gg2 + theme(panel.grid.major = element_blank(),
                     panel.border = element_blank(),
                     axis.text.x = element_blank(),
                     axis.ticks = element_blank())
}

#print info's
print (verisplot2(vcdb_verizon, "action"))
```



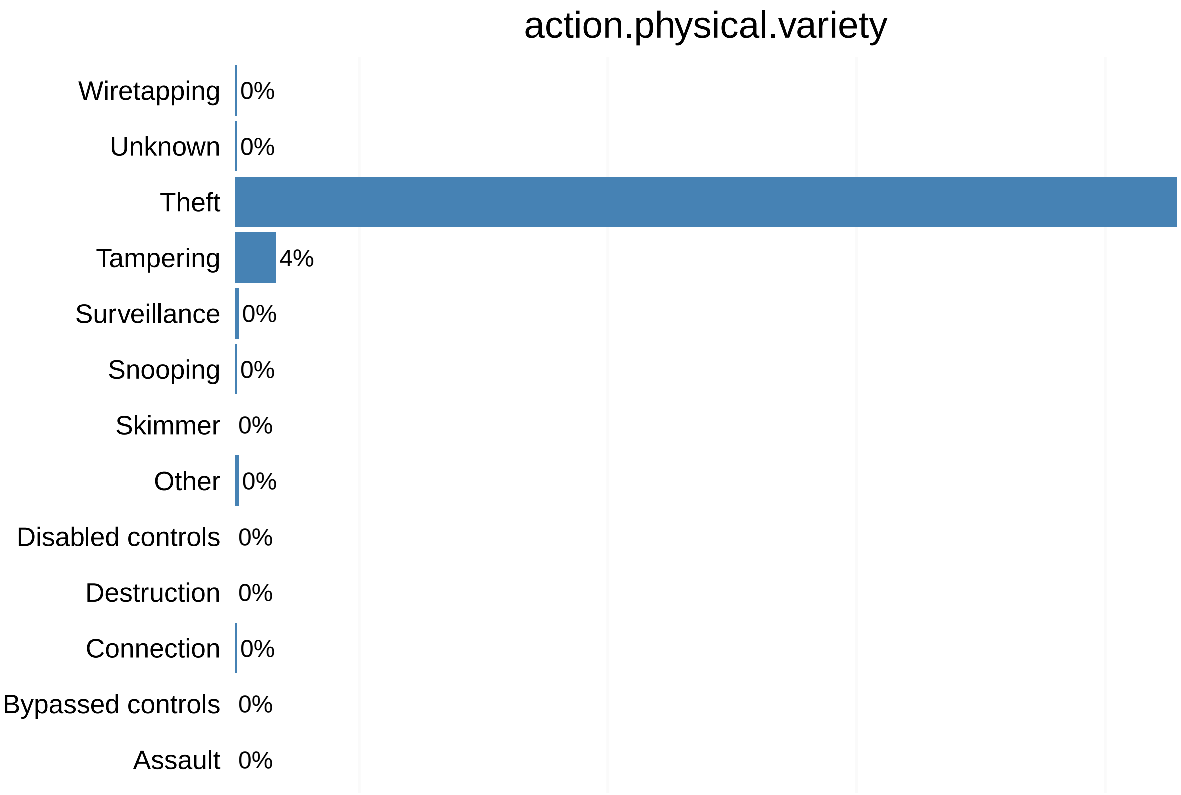
```
print(verisplot(vcdb_orginal, "actor.external.variety"))
```



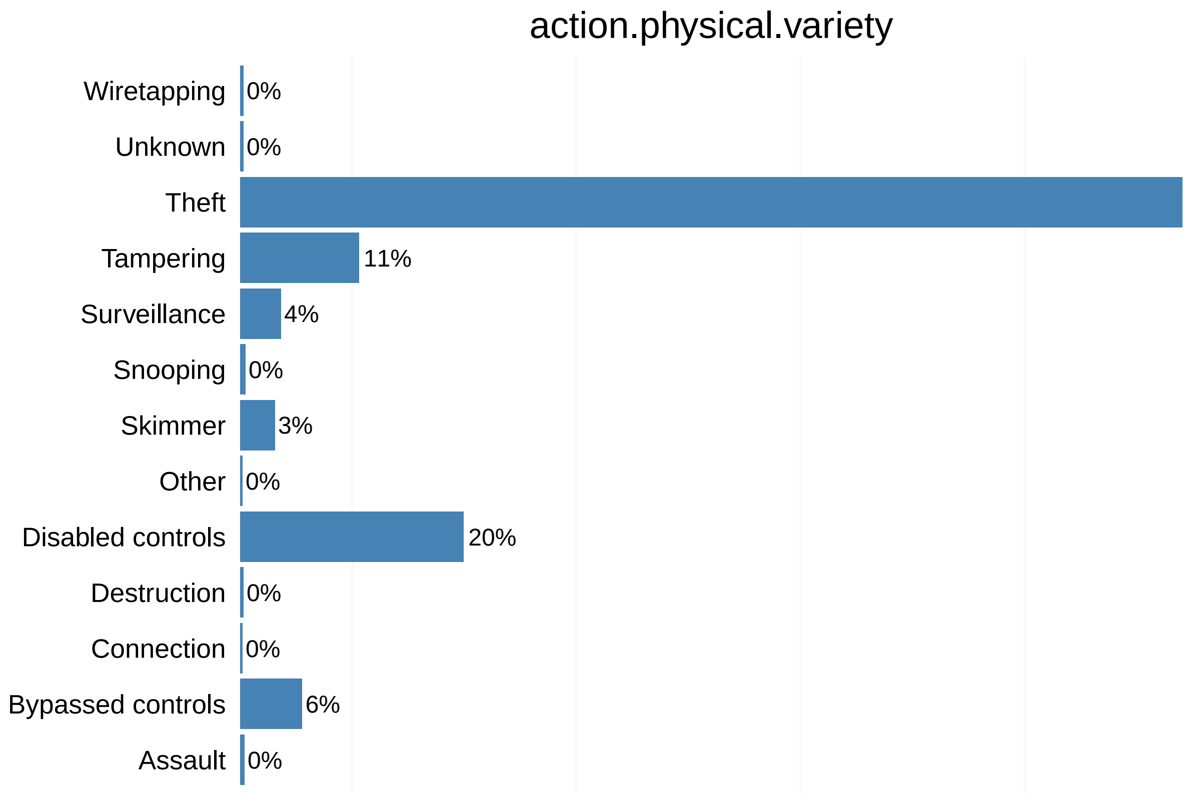
```r
print(verisplot2(vcdb_verizon, "actor.external.variety"))
```



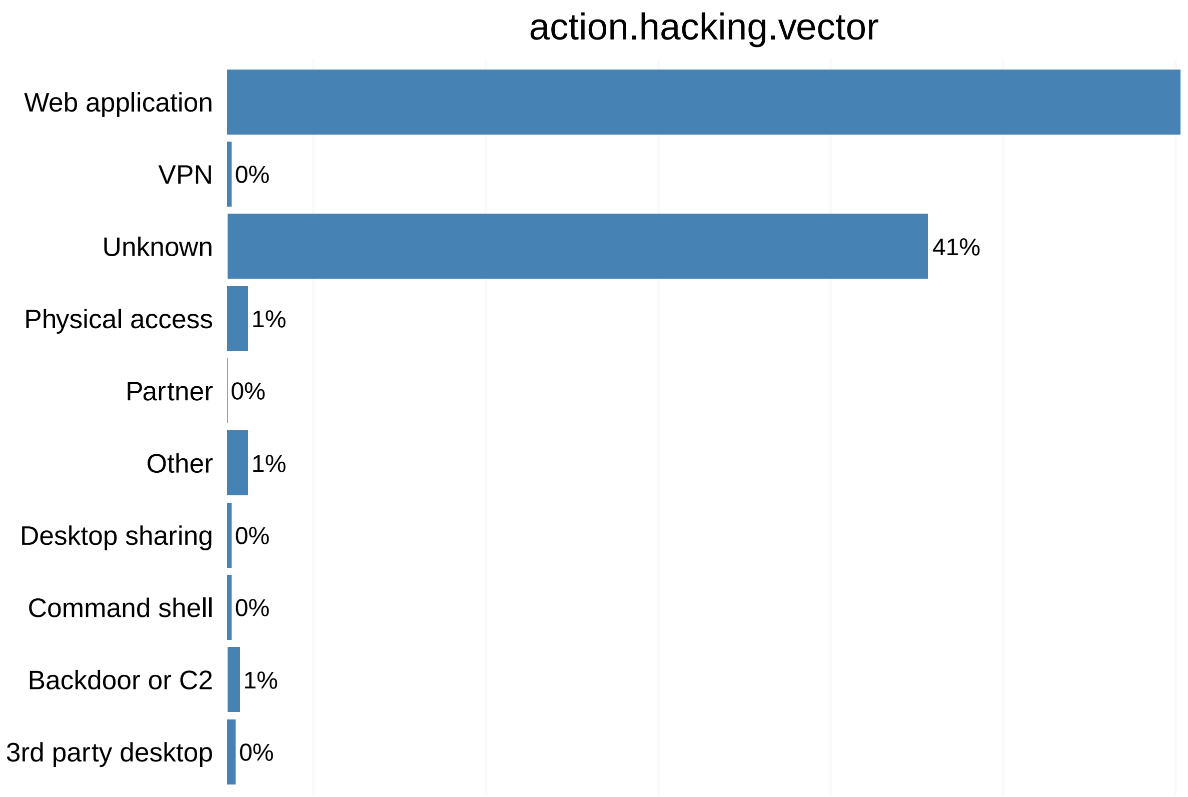
```
print(verisplot(vcdb_orginal, "action.physical.variety"))
```



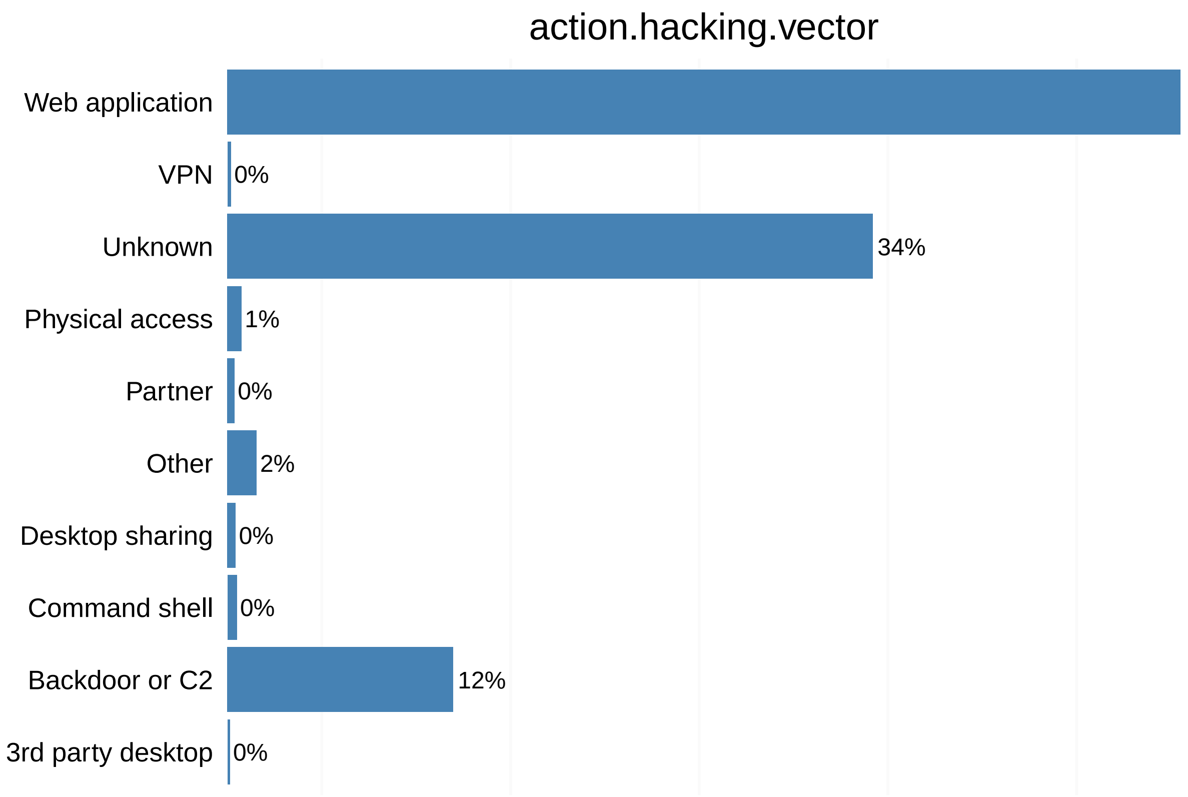
```
print(verisplot2(vcdb_verizon, "action.physical.variety"))
```



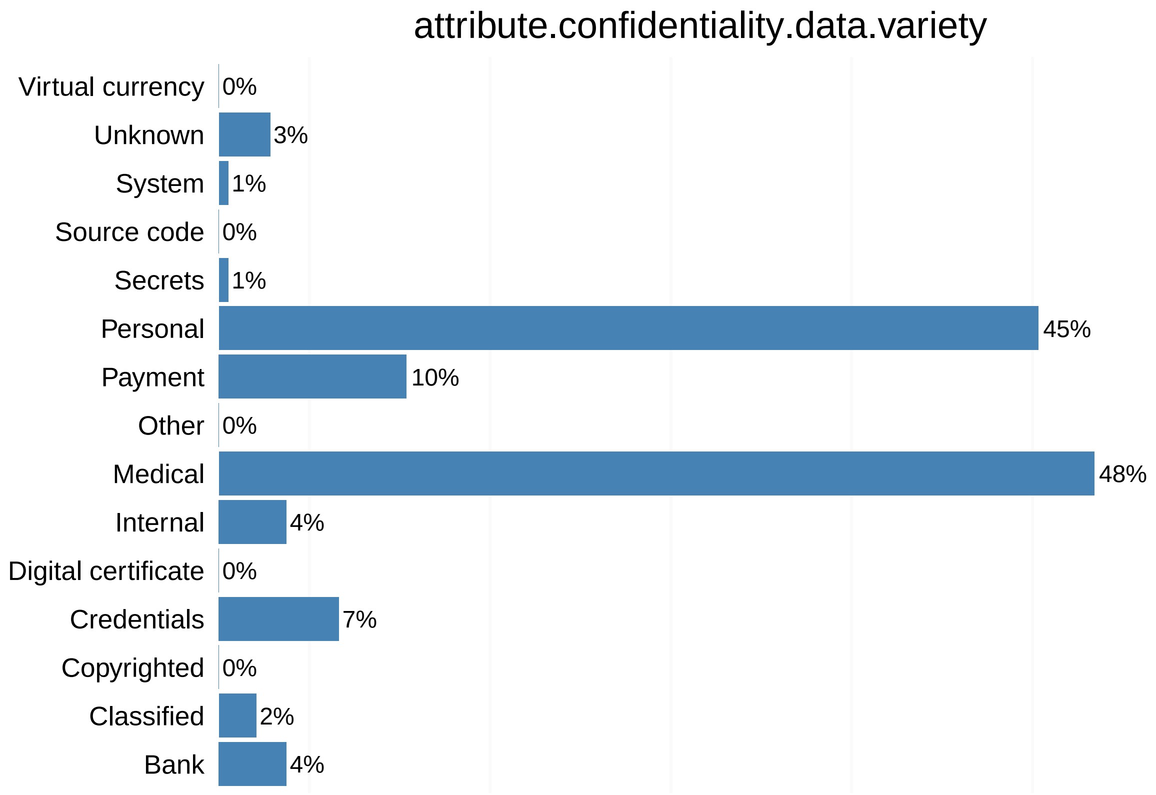
```
print(verisplot(vcdb_orginal, "action.hacking.vector"))
```



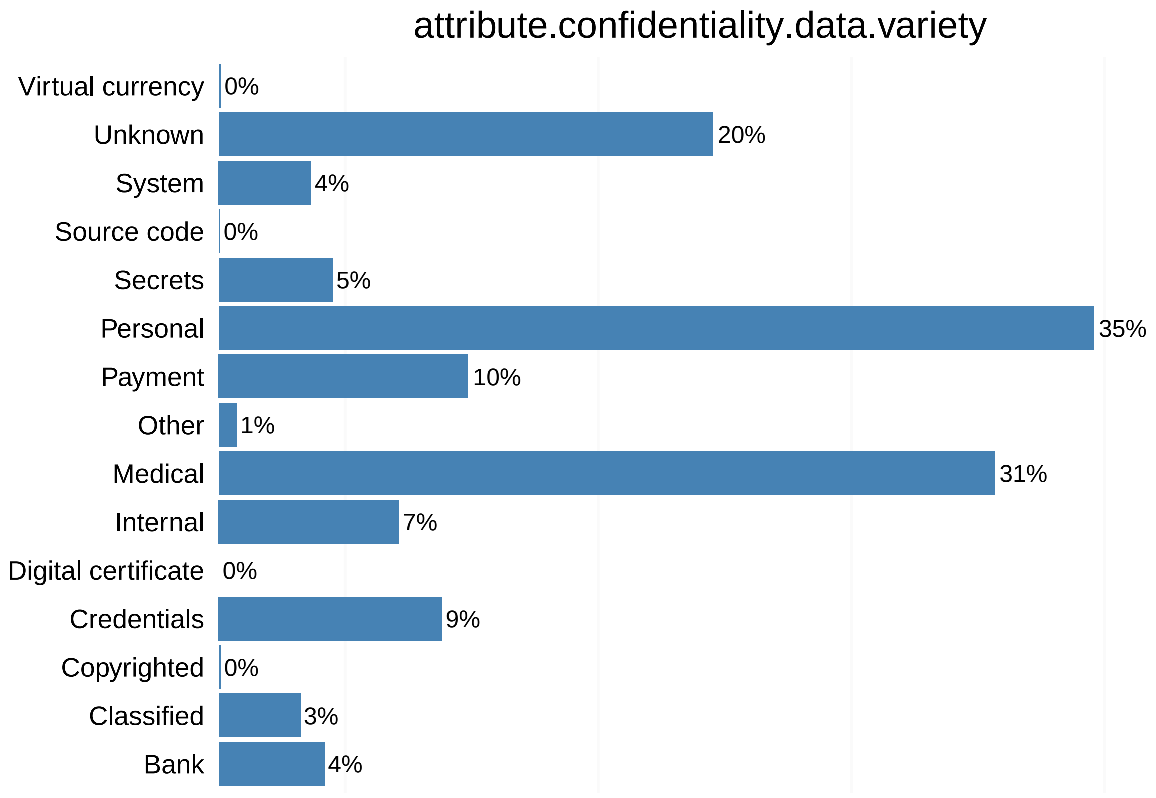
```
print(verisplot2(vcdb_verizon, "action.hacking.vector"))
```



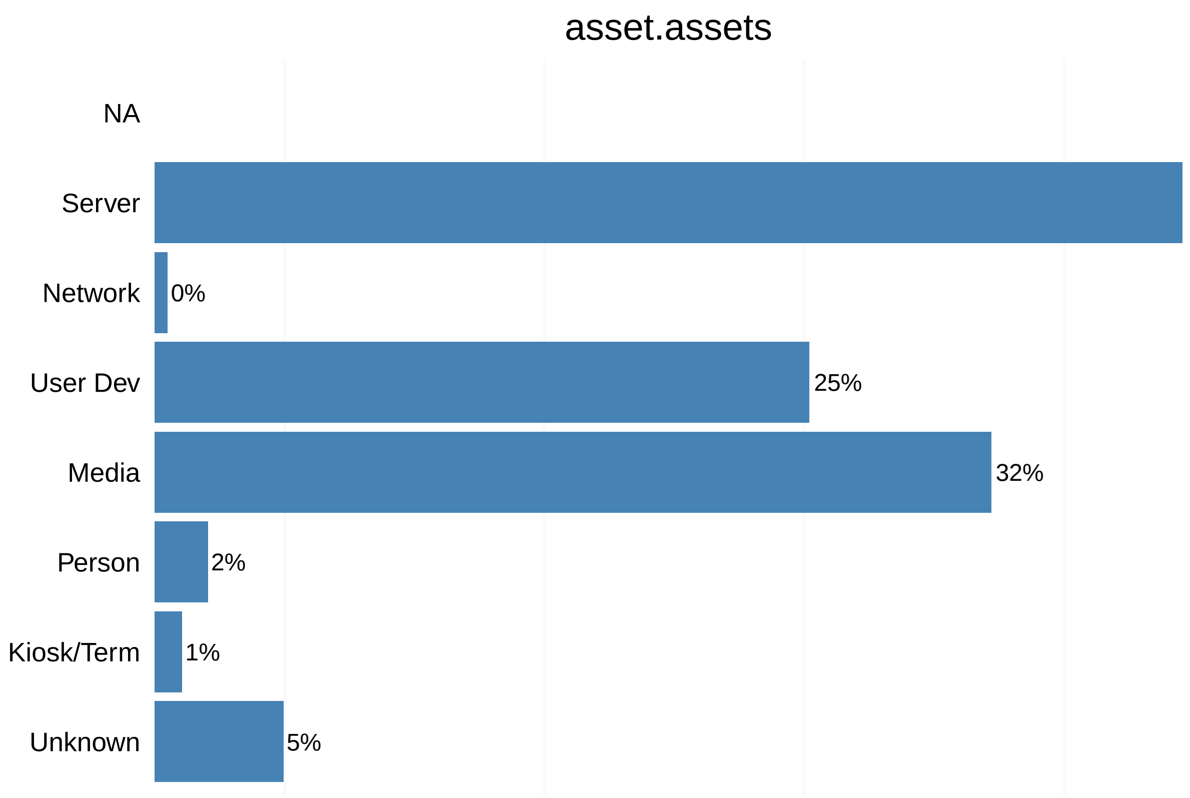
```
print(verisplot(vcdb_orginal, "attribute.confidentiality.data.variety"))
```



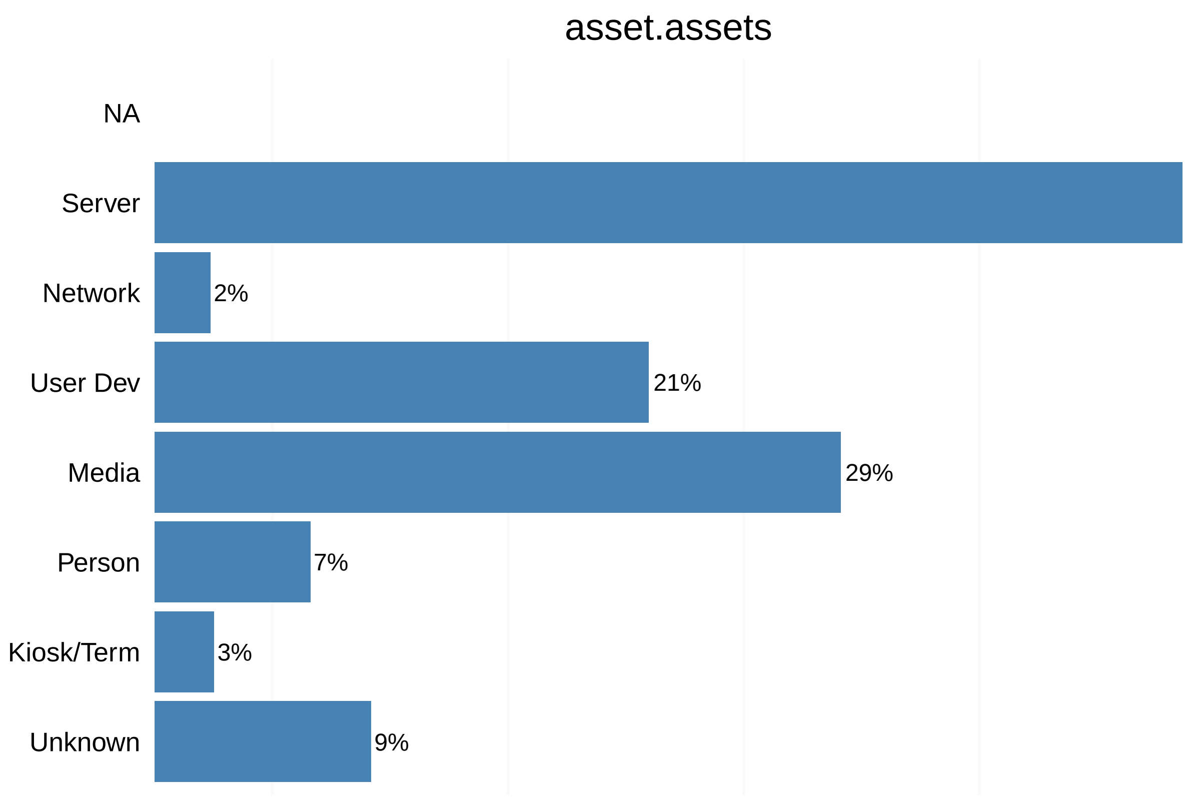
```r
print(verisplot2(vcdb_verizon, "attribute.confidentiality.data.variety"))
```



```
print(verisplot(vcdb_orginal, "asset.assets"))
```



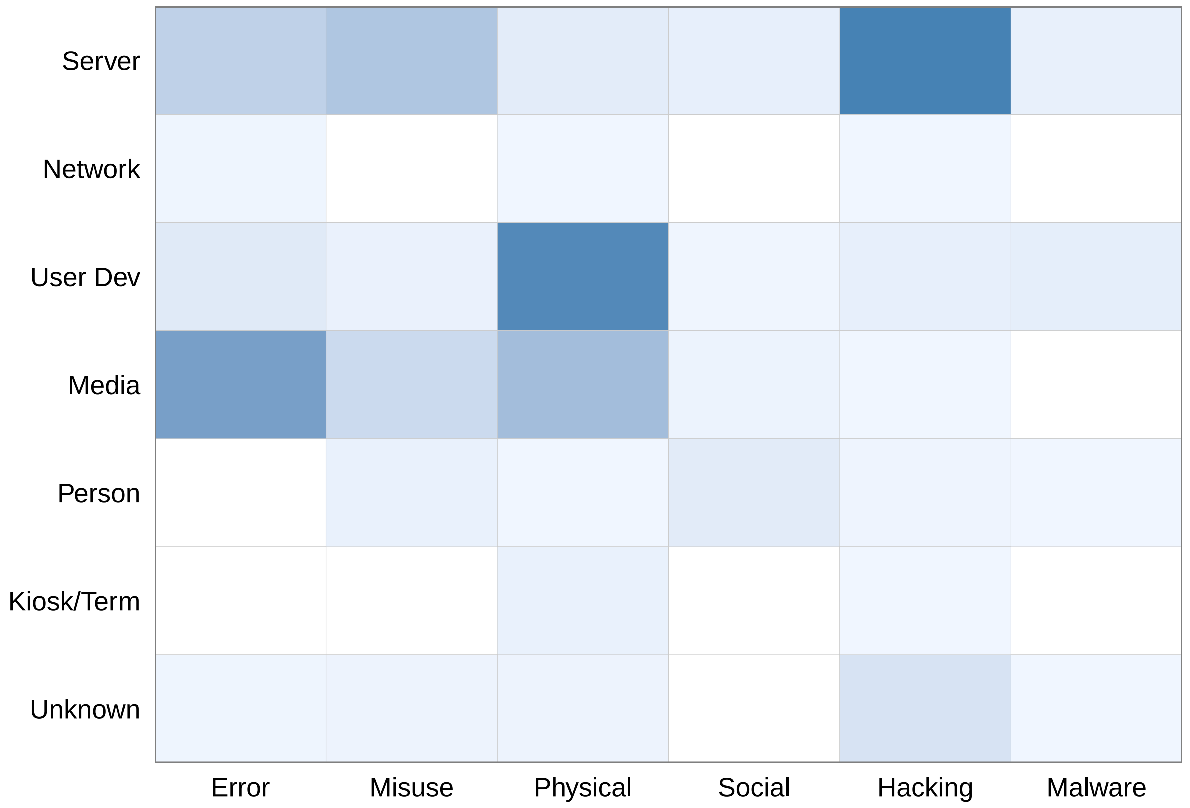
```r
print(verisplot2(vcdb_verizon, "asset.assets"))
```



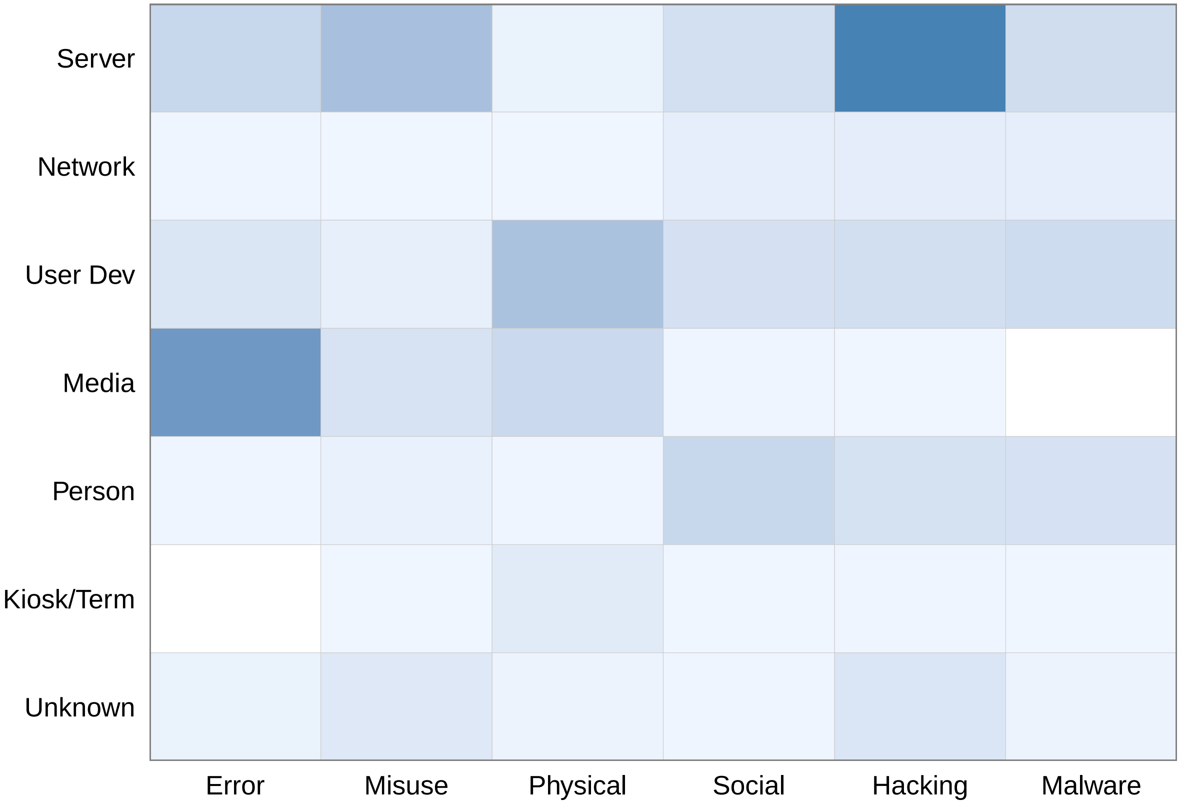
```r
a2 <- getenum(vcdb_orginal, enum="action", primary="asset.assets", add.freq=T)
a2
```

```
##             enum     enum1   x    n         freq
##  1:      Malware    Server  18 1630 0.0110429448
##  2:      Hacking    Server 339 1630 0.2079754601
##  3:       Social    Server  20 1630 0.0122699387
##  4:     Physical    Server  29 1630 0.0177914110
##  5:       Misuse    Server 137 1630 0.0840490798
##  6:        Error    Server 104 1630 0.0638036810
##  7: Environmental   Server   0 1630 0.0000000000
##  8:      Unknown    Server   0 1630 0.0000000000
##  9:      Malware   Network   0 1630 0.0000000000
## 10:       Hacking    Network   2 1630 0.0012269939
## 11:        Social    Network   0 1630 0.0000000000
## 12:      Physical    Network   1 1630 0.0006134969
## 13:        Misuse    Network   0 1630 0.0000000000
## 14:         Error    Network   5 1630 0.0030674847
## 15: Environmental    Network   0 1630 0.0000000000
## 16:       Unknown    Network   0 1630 0.0000000000
## 17:       Malware   User Dev  23 1630 0.0141104294
## 18:       Hacking   User Dev  20 1630 0.0122699387
## 19:        Social   User Dev   4 1630 0.0024539877
## 20:      Physical   User Dev 317 1630 0.1944785276
## 21:        Misuse   User Dev  14 1630 0.0085889571
## 22:         Error   User Dev  35 1630 0.0214723926
## 23: Environmental   User Dev   0 1630 0.0000000000
## 24:       Unknown   User Dev   0 1630 0.0000000000
## 25:       Malware      Media   0 1630 0.0000000000
## 26:       Hacking      Media   2 1630 0.0012269939
## 27:        Social      Media  10 1630 0.0061349693
## 28:      Physical      Media 163 1630 0.1000000000
## 29:        Misuse      Media  78 1630 0.0478527607
## 30:         Error      Media 250 1630 0.1533742331
## 31: Environmental      Media   0 1630 0.0000000000
## 32:       Unknown      Media   0 1630 0.0000000000
## 33:       Malware     Person   1 1630 0.0006134969
## 34:       Hacking     Person   6 1630 0.0036809816
## 35:        Social     Person  31 1630 0.0190184049
## 36:      Physical     Person   2 1630 0.0012269939
## 37:        Misuse     Person  15 1630 0.0092024540
## 38:         Error     Person   0 1630 0.0000000000
## 39: Environmental     Person   0 1630 0.0000000000
## 40:       Unknown     Person   0 1630 0.0000000000
## 41:       Malware Kiosk/Term   0 1630 0.0000000000
## 42:       Hacking Kiosk/Term   1 1630 0.0006134969
## 43:        Social Kiosk/Term   0 1630 0.0000000000
## 44:      Physical Kiosk/Term  16 1630 0.0098159509
## 45:        Misuse Kiosk/Term   0 1630 0.0000000000
## 46:         Error Kiosk/Term   0 1630 0.0000000000
## 47: Environmental Kiosk/Term   0 1630 0.0000000000
## 48:       Unknown Kiosk/Term   0 1630 0.0000000000
## 49:       Malware    Unknown   1 1630 0.0006134969
## 50:       Hacking    Unknown  53 1630 0.0325153374
## 51:        Social    Unknown   0 1630 0.0000000000
## 52:      Physical    Unknown   8 1630 0.0049079755
## 53:        Misuse    Unknown   8 1630 0.0049079755
## 54:         Error    Unknown   5 1630 0.0030674847
## 55: Environmental    Unknown   0 1630 0.0000000000
## 56:       Unknown    Unknown   0 1630 0.0000000000
##              enum      enum1  x    n         freq
```

```
# trim useless data
a2 <- a2[which(a2$enum!="Environmental" & a2$enum!="Unknown"), ]
#make slim version
slim.a2 <- a2[which(a2$x!=0), ]
# plot df
gg <- ggplot(a2, aes(x=enum, y=enum1, fill=freq))
gg <- gg + geom_tile(fill="white", color="gray80")
gg <- gg + geom_tile(data=slim.a2, color="gray80")
gg <- gg + scale_fill_gradient(low = "#F0F6FF",
                               high = "#4682B4", guide=F)
gg <- gg + xlab("") + ylab("") + theme_bw()
gg <- gg + scale_x_discrete(expand=c(0,0))
gg <- gg + scale_y_discrete(expand=c(0,0))
gg <- gg + theme(axis.ticks = element_blank())
# display plot
print(gg)
```



```
a22 <- getenum(vcdb_verizon, enum="action", primary="asset.assets", add.freq=T)
# remove all unknow and enviromental in df
a22 <- a22[which(a22$enum!="Environmental" & a22$enum!="Unknown"), ]
# slim version
slim.a22 <- a22[which(a22$x!=0), ]
# could be sorting
# make plot
gg <- ggplot(a22, aes(x=enum, y=enum1, fill=freq))
gg <- gg + geom_tile(fill="white", color="gray80")
gg <- gg + geom_tile(data=slim.a22, color="gray80")
gg <- gg + scale_fill_gradient(low = "#F0F6FF",
                               high = "#4682B4", guide=F)
gg <- gg + xlab("") + ylab("") + theme_bw()
gg <- gg + scale_x_discrete(expand=c(0,0))
gg <- gg + scale_y_discrete(expand=c(0,0))
gg <- gg + theme(axis.ticks = element_blank())
# and view it
print(gg)
```


Server
Network
User Dev
Media
Person
Kiosk/Term
Unknown
Error
Misuse
Physical
Social
Hacking
Malware