

Cyber Research

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.3      v purrr   0.3.4
## v tibble  3.1.1      v dplyr   1.0.5
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   1.4.0      v forcats 0.5.1
```

```
## Warning: package 'tibble' was built under R version 4.0.5
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'readr' was built under R version 4.0.5
```

```
## Warning: package 'purrr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## Warning: package 'forcats' was built under R version 4.0.5
```

```
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(dplyr)
library(ggplot2)
library(fastDummies)
```

```
## Warning: package 'fastDummies' was built under R version 4.0.5
```

```
#install.packages("patchwork")
library(patchwork)
```

```
## Warning: package 'patchwork' was built under R version 4.0.5
```

```
df.edu.safety = read.csv("https://raw.githubusercontent.com/tahlla-utd/cybersecresearch/main/CyberResearchData/edu.safety.csv",
                          header = TRUE, sep = ';')
```

```
#identifying the rows with NAs
```

```
rownames(df.edu.safety)[apply(df.edu.safety, 2, anyNA)]
```

```
## [1] "18" "19" "20" "21" "22" "46" "47" "48" "49" "50" "74" "75"  
## [13] "76" "77" "78" "102" "103" "104" "105" "106" "130" "131" "132" "133"  
## [25] "134" "158" "159" "160" "161" "162" "186" "187" "188" "189" "190" "214"  
## [37] "215" "216" "217" "218" "242" "243" "244" "245" "246" "270" "271" "272"  
## [49] "273" "274" "298" "299" "300" "301" "302" "326" "327" "328" "329" "330"  
## [61] "354" "355" "356" "357" "358" "382" "383" "384" "385" "386" "410" "411"  
## [73] "412" "413" "414" "438" "439" "440" "441" "442" "466" "467" "468" "469"  
## [85] "470" "494" "495" "496" "497" "498" "522" "523" "524" "525" "526" "550"  
## [97] "551" "552" "553" "554" "578" "579" "580" "581" "582"
```

```
#removing all observations with NAs
```

```
df.clean <- df.edu.safety %>% na.omit()
```

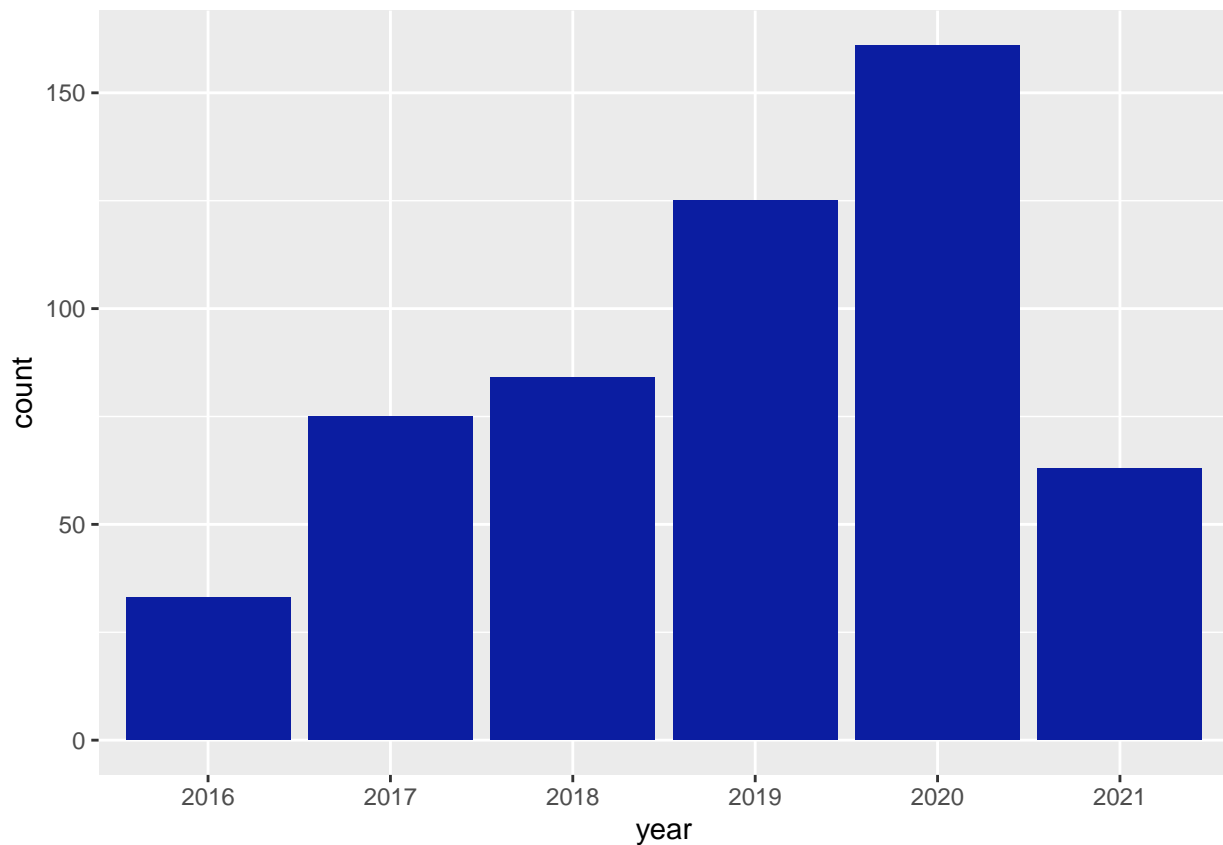
```
df.clean <- df.clean
```

```
df.clean$year = as.factor(df.clean$year)
```

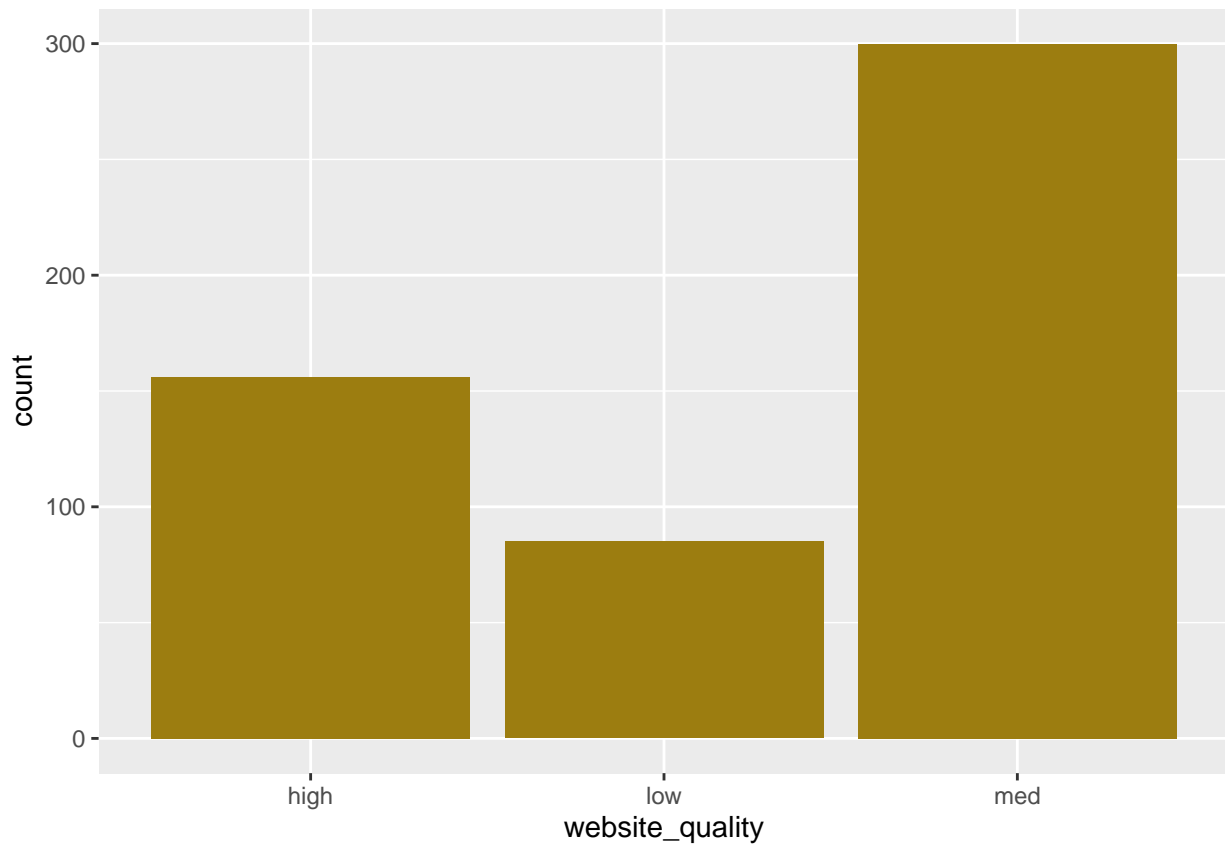
Linear Regression of Various variables

```
test.plot <- ggplot(data = df.clean, aes(x = year))+geom_bar(fill = "#0b1da1")
```

```
test.plot
```



```
test.plot <- ggplot(data = df.clean, aes(x = website_quality))+geom_bar(fill = "#9c7d10")
test.plot
```



```
#create data frame of attack type counts
```

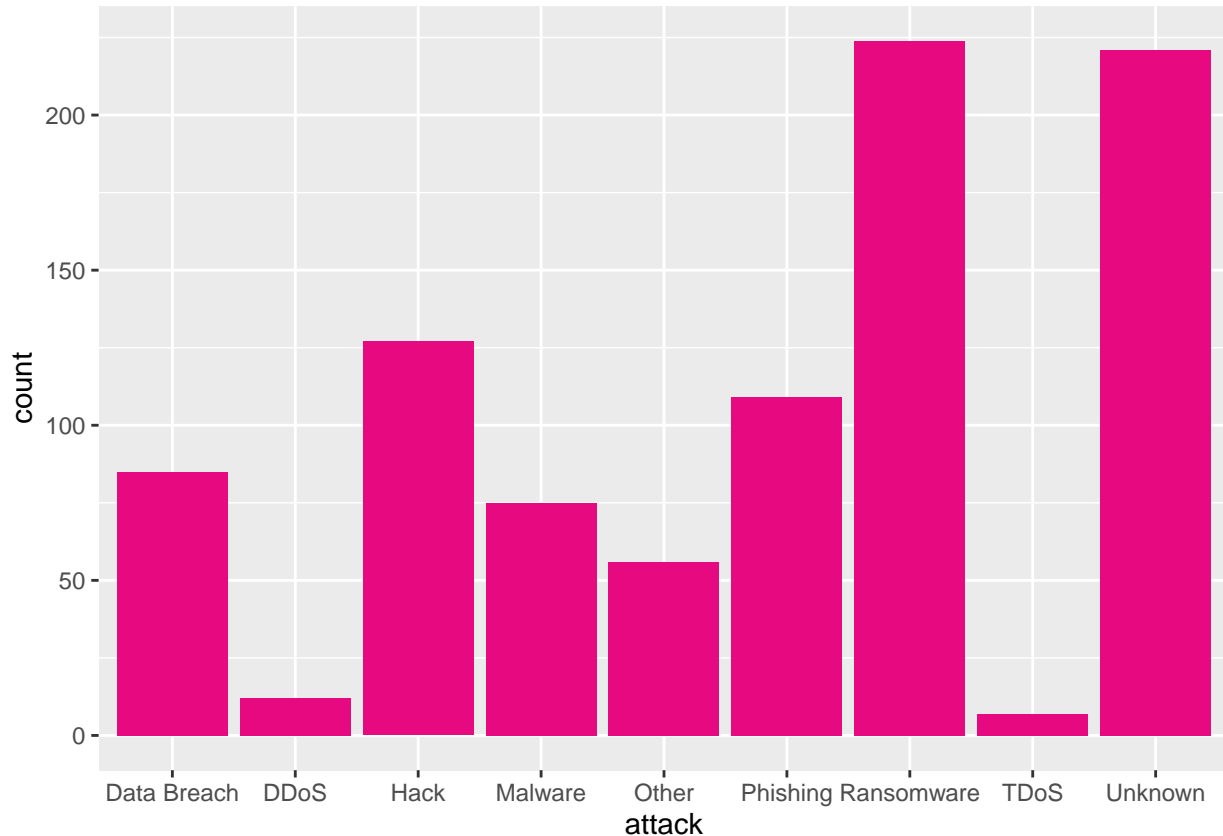
```
df.attack.count <- data.frame(matrix(ncol = 9, nrow = 1))
x <- c("Unknown", "Hack", "Data Breach", "Ransomware", "Phishing", "TDoS", "Malware", "DDoS", "Other")
colnames(df.attack.count) <- x
```

```
df.attack.count[1, "Unknown"] <- sum(df.clean$Unknown)
df.attack.count[1, "Hack"] <- sum(df.clean$Hack)
df.attack.count[1, "Data Breach"] <- sum(df.clean$Data_Breach)
df.attack.count[1, "Ransomware"] <- sum(df.clean$Ransomware)
df.attack.count[1, "Phishing"] <- sum(df.clean$Phishing)
df.attack.count[1, "TDoS"] <- sum(df.clean$TDoS)
df.attack.count[1, "Malware"] <- sum(df.clean$Malware)
df.attack.count[1, "DDoS"] <- sum(df.clean$DDoS)
df.attack.count[1, "Other"] <- sum(df.clean$Other)
```

```
head(df.attack.count)
```

```
##   Unknown Hack Data Breach Ransomware Phishing TDoS Malware DDoS Other
## 1    221  127          85         224    109    7      75   12   56
```

```
df.long <- df.attack.count %>%
  pivot_longer(Unknown:Other, names_to = "attack", values_to = "count")
ggplot(df.long, aes(x = attack, y = count)) +
  geom_col(fill = "#e6097f")
```



```
graph.attack.by.filter <- function(fill.choice){
  df.unk <- df.clean %>% as_tibble() %>%
    mutate(Unknown = as.factor(if_else(Unknown == 1, "True", "False")))

  unk.graph <- ggplot(data = df.unk, aes(Unknown == "True", fill = fill.choice))+geom_bar(position = "dodge")
  #####

  df.hack <- df.clean %>% as_tibble() %>%
    mutate(Hack = as.factor(if_else(Hack == 1, "True", "False")))

  hack.graph <- ggplot(data = df.hack, aes(Hack == "True", fill = fill.choice))+geom_bar(position = "dodge")
  #####

  df.data <- df.clean %>% as_tibble() %>%
    mutate(Data_Breach = as.factor(if_else(Data_Breach == 1, "True", "False")))

  data.graph <- ggplot(data = df.data, aes(Data_Breach == "True", fill = fill.choice))+geom_bar(position = "dodge")
  #####

  df.ran <- df.clean %>% as_tibble() %>%
    mutate(Ransomware = as.factor(if_else(Ransomware == 1, "True", "False")))

  ran.graph <- ggplot(data = df.ran, aes(Ransomware == "True", fill = fill.choice))+geom_bar(position = "dodge")
  #####
```

```
#####
df.phish <- df.clean %>% as_tibble() %>%
  mutate(Phishing = as.factor(if_else(Phishing == 1, "True", "False")))

phish.graph <- ggplot(data = df.phish, aes(Phishing == "True", fill = fill.choice))+geom_bar(position =
#####
df.tdos <- df.clean %>% as_tibble() %>%
  mutate(TDoS = as.factor(if_else(TDoS == 1, "True", "False")))

tdos.graph <- ggplot(data = df.tdos, aes(TDoS == "True", fill = fill.choice))+geom_bar(position = "dodge
#####
df.mal <- df.clean %>% as_tibble() %>%
  mutate(Malware = as.factor(if_else(Malware == 1, "True", "False")))

mal.graph <- ggplot(data = df.mal, aes(Malware == "True", fill = fill.choice))+geom_bar(position = "dodge
#####
df.ddos <- df.clean %>% as_tibble() %>%
  mutate(DDoS = as.factor(if_else(DDoS == 1, "True", "False")))

ddos.graph <- ggplot(data = df.ddos, aes(DDoS == "True", fill = fill.choice))+geom_bar(position = "dodge
#####
df.other <- df.clean %>% as_tibble() %>%
  mutate(Other = as.factor(if_else(Other == 1, "True", "False")))

other.graph <- ggplot(data = df.other, aes(Other == "True", fill = fill.choice))+geom_bar(position = "dodge
#####
df.serv <- df.clean %>% as_tibble() %>%
  mutate(Server_Shutdown = as.factor(if_else(Server_Shutdown == 1, "True", "False")))

serv.graph <- ggplot(data = df.serv, aes(Server_Shutdown == "True", fill = fill.choice))+geom_bar(position = "dodge
#####

(data.graph + scale_x_discrete(limit = c(TRUE)) | ddos.graph + scale_x_discrete(limit = c(TRUE)) |
  hack.graph + scale_x_discrete(limit = c(TRUE)))/ (mal.graph + scale_x_discrete(limit = c(TRUE)) |
  other.graph + scale_x_discrete(limit = c(TRUE)) | phish.graph + scale_x_discrete(limit = c(TRUE)))/
  (ran.graph + scale_x_discrete(limit = c(TRUE)) | tdos.graph + scale_x_discrete(limit = c(TRUE)) | unk
  scale_x_discrete(limit = c(TRUE)))
}
```

```
graph.attack.by.filter(df.clean$year)
```

```
## Warning: Removed 456 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 529 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 414 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 466 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 485 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 432 rows containing non-finite values (stat_count).
## Warning: Removed 317 rows containing non-finite values (stat_count).
## Warning: Removed 534 rows containing non-finite values (stat_count).
## Warning: Removed 320 rows containing non-finite values (stat_count).
```



```
graph.attack.by.filter(df.clean$type)
```

```
## Warning: Removed 456 rows containing non-finite values (stat_count).
## Warning: Removed 529 rows containing non-finite values (stat_count).
## Warning: Removed 414 rows containing non-finite values (stat_count).
## Warning: Removed 466 rows containing non-finite values (stat_count).
## Warning: Removed 485 rows containing non-finite values (stat_count).
## Warning: Removed 432 rows containing non-finite values (stat_count).
## Warning: Removed 317 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 534 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 320 rows containing non-finite values (stat_count).
```



```
graph.attack.by.filter((df.clean$website_quality))
```

```
## Warning: Removed 456 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 529 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 414 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 466 rows containing non-finite values (stat_count).
```

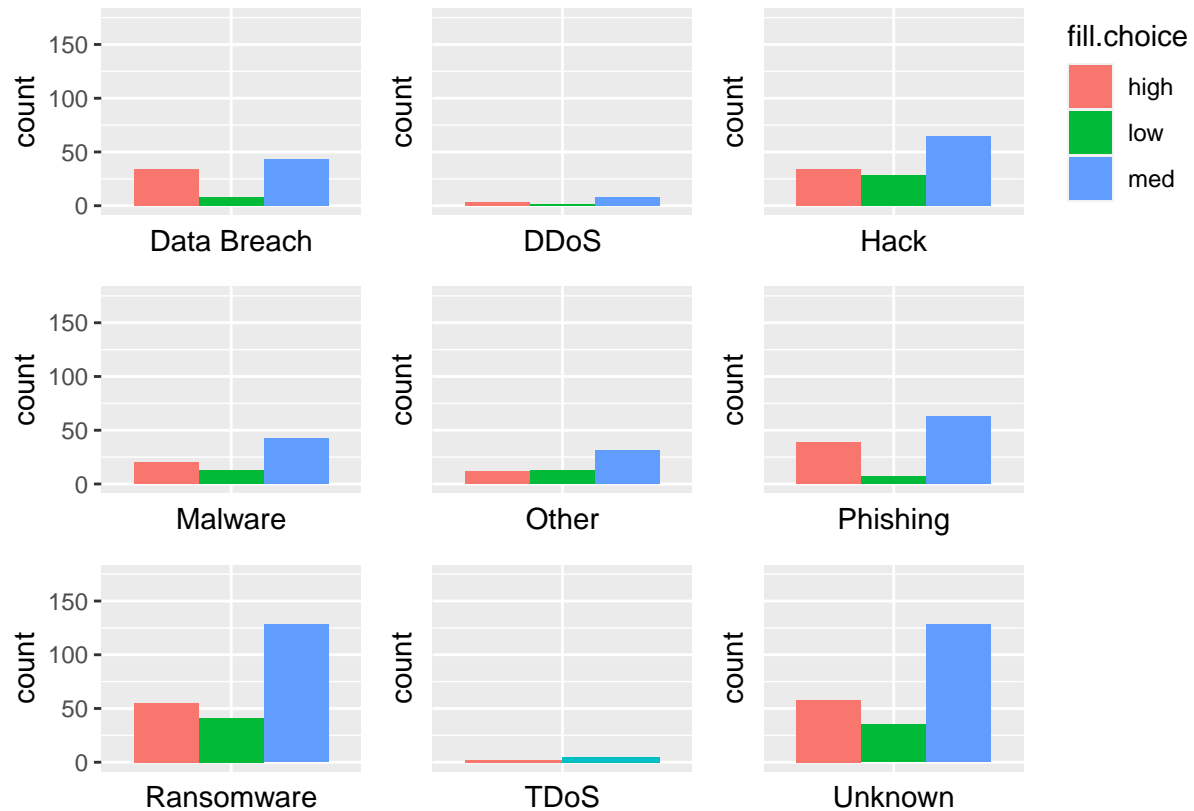
```
## Warning: Removed 485 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 432 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 317 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 534 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 320 rows containing non-finite values (stat_count).
```



```
totalattacks <- function(year){
  totalsum = 0;
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "Unknown"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "Data_Breach"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "Malware"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "TDoS"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "DDoS"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "Ransomware"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "Phishing"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "Other"]));
  totalsum = totalsum + sum((df.clean[which(df.clean$year == year), "Hack"]));

  return(totalsum)
}
```

```
year.trend <-function(attack.type, color.choice){
```

```
  year.list <- c("2016", "2017", "2018", "2019", "2020", "2021")
  df.tmp <- data.frame(matrix(ncol = 6, nrow = 1))
  colnames(df.tmp) <- year.list
```

```
  df.tmp[1, "2016"] <- sum((df.clean[which(df.clean$year == "2016"), attack.type])/totalattacks("2016"))
  df.tmp[1, "2017"] <- sum((df.clean[which(df.clean$year == "2017"), attack.type])/totalattacks("2017"))
  df.tmp[1, "2018"] <- sum((df.clean[which(df.clean$year == "2018"), attack.type])/totalattacks("2018"))
  df.tmp[1, "2019"] <- sum((df.clean[which(df.clean$year == "2019"), attack.type])/totalattacks("2019"))
```



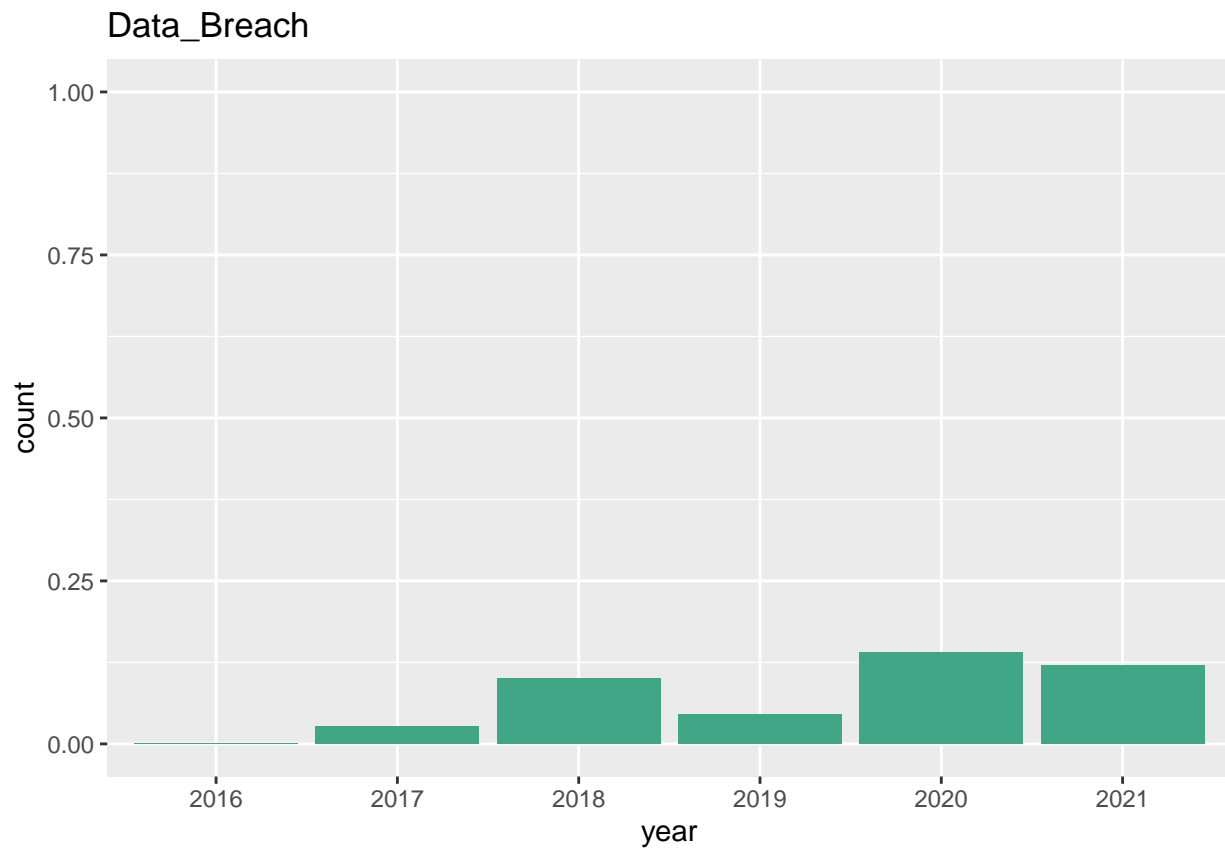
```

df.tmp[1, "2020"] <- sum((df.clean[which(df.clean$year == "2020"), attack.type])/totalattacks("2020"))
df.tmp[1, "2021"] <- sum((df.clean[which(df.clean$year == "2021"), attack.type])/totalattacks("2021"))

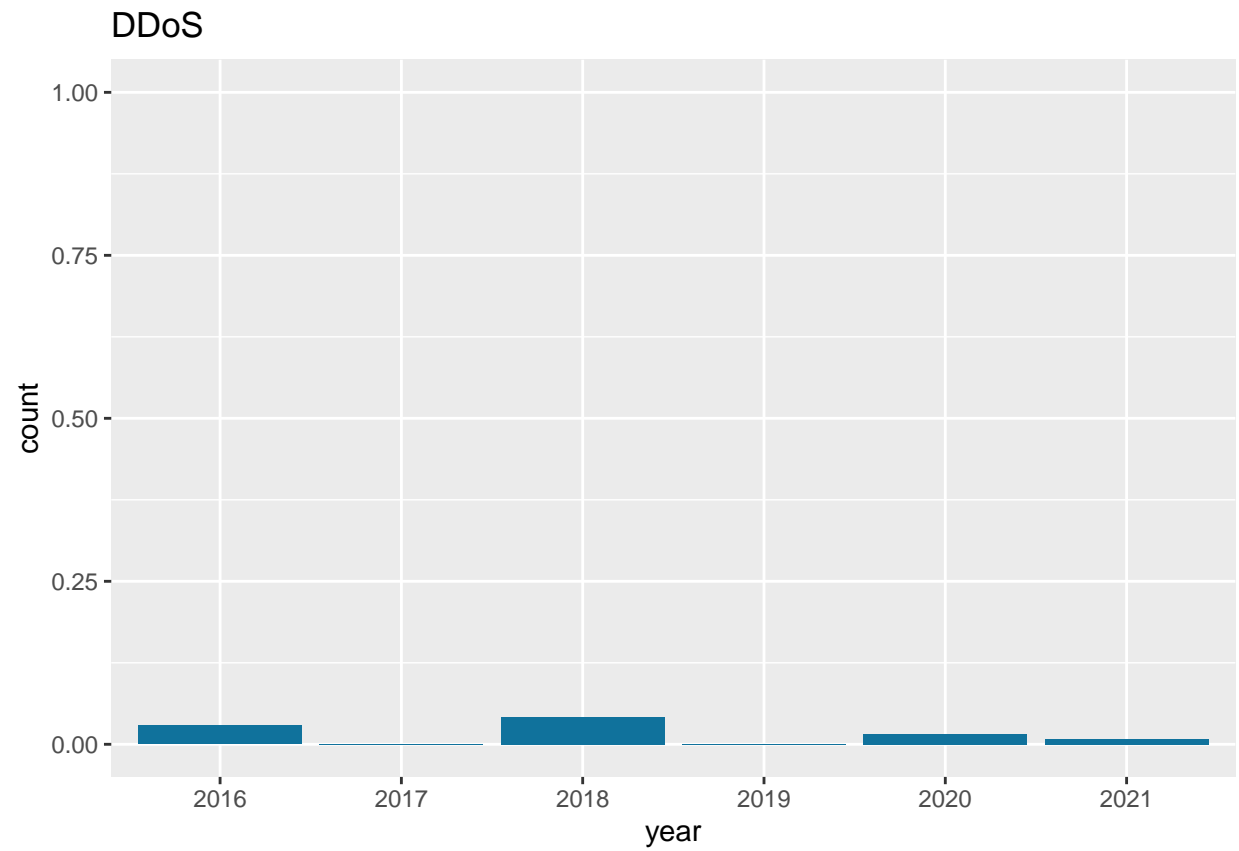
df.long <- df.tmp %>%
  pivot_longer(1:6, names_to = "year", values_to = "count")
ggplot(df.long, aes(x = year, y = count)) +
  geom_col(fill = color.choice) + labs(title = attack.type) + ylim(0, 1)
}

```

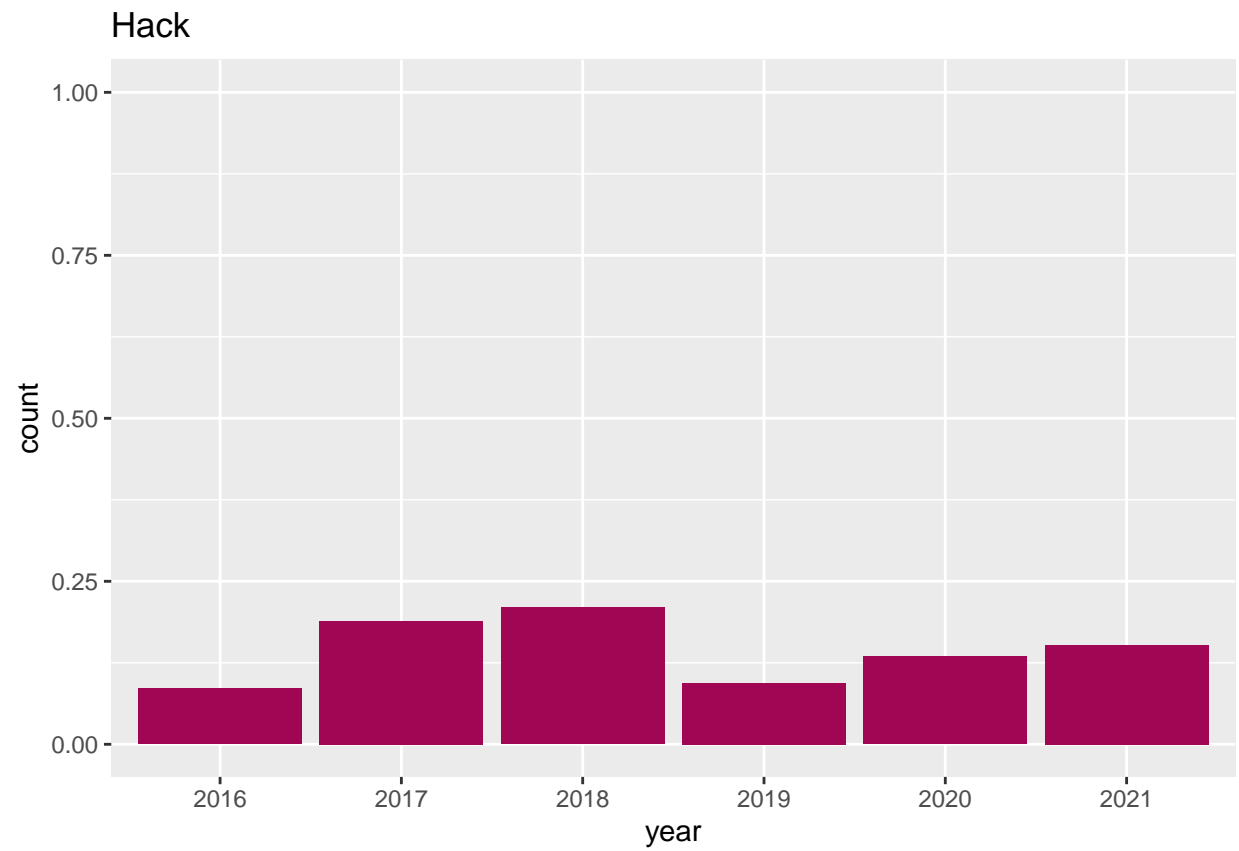
```
year.trend("Data_Breach", "#41a686")
```



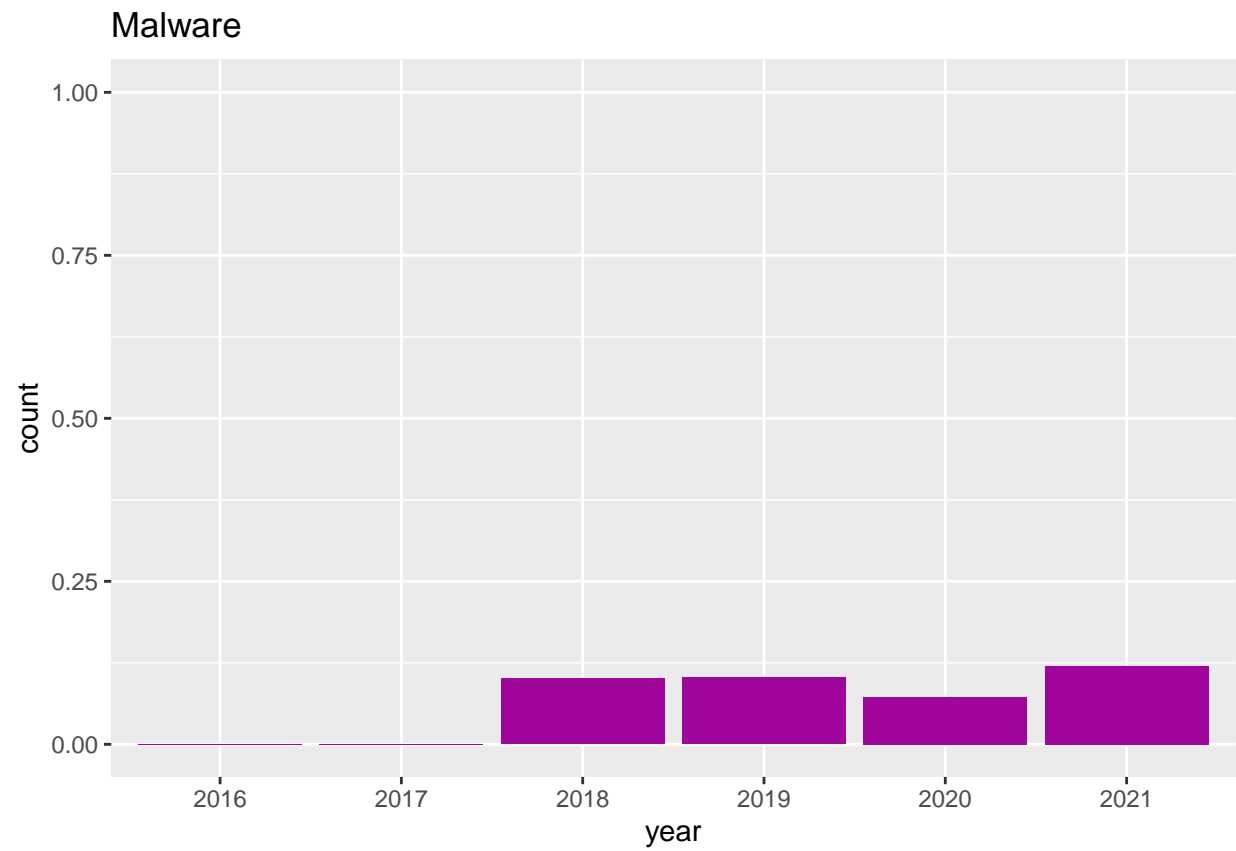
```
year.trend("DDoS", "#10729c")
```



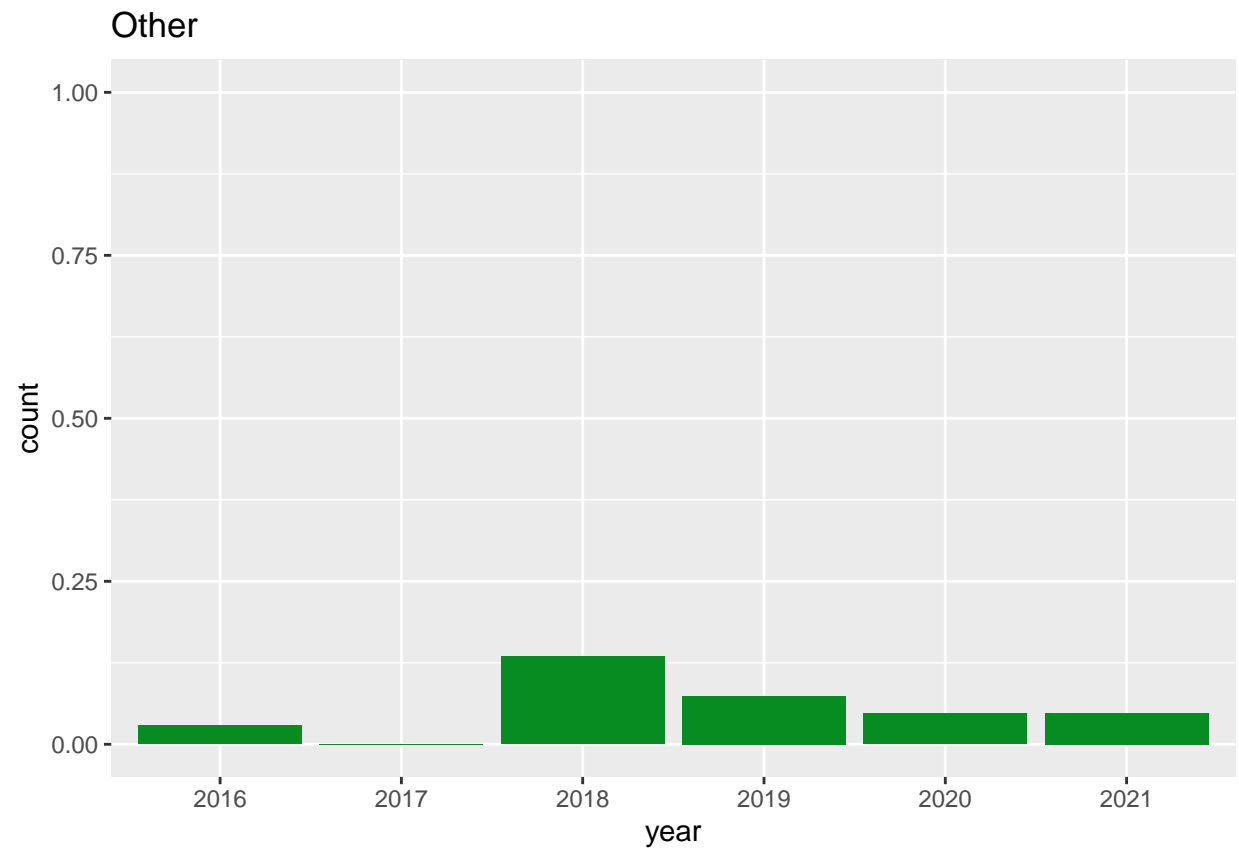
```
year.trend("Hack", "#a10654")
```



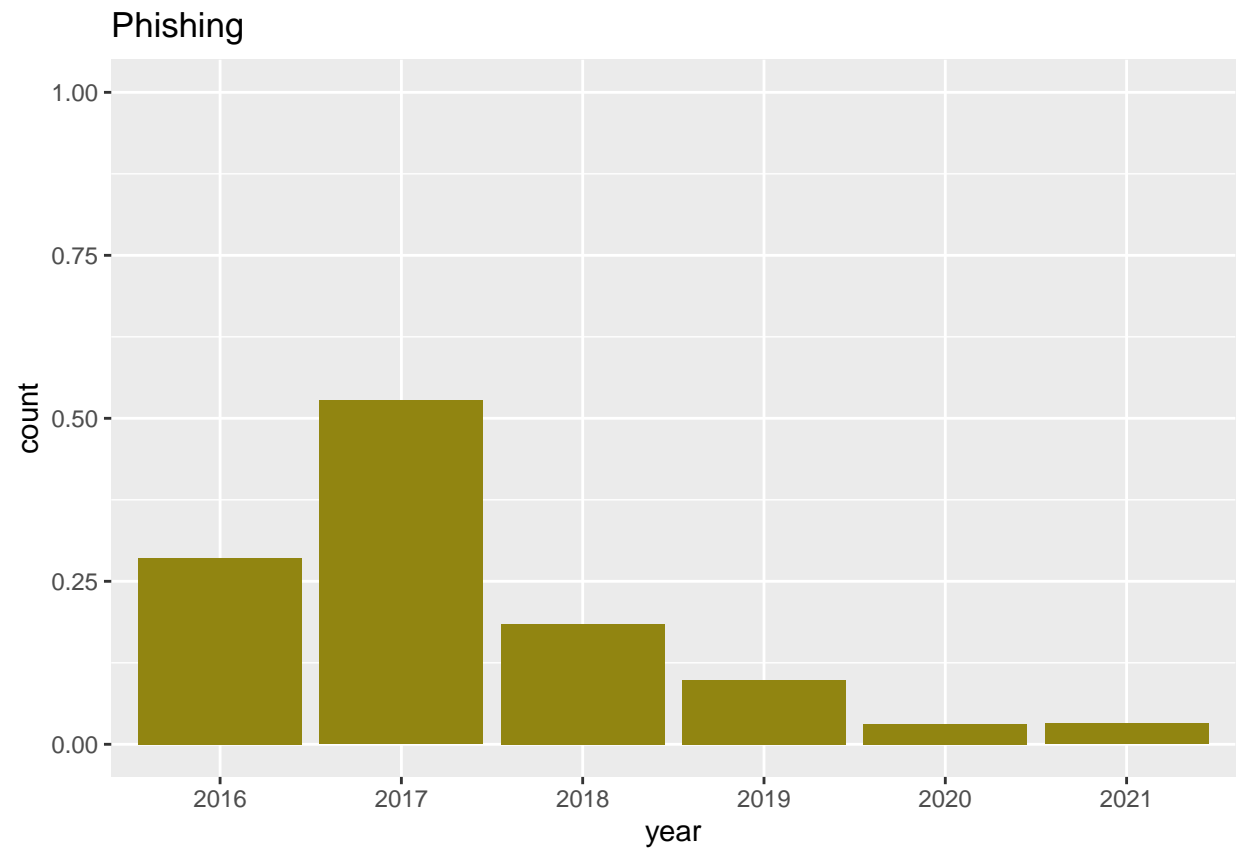
```
year.trend("Malware", "#a1069c")
```



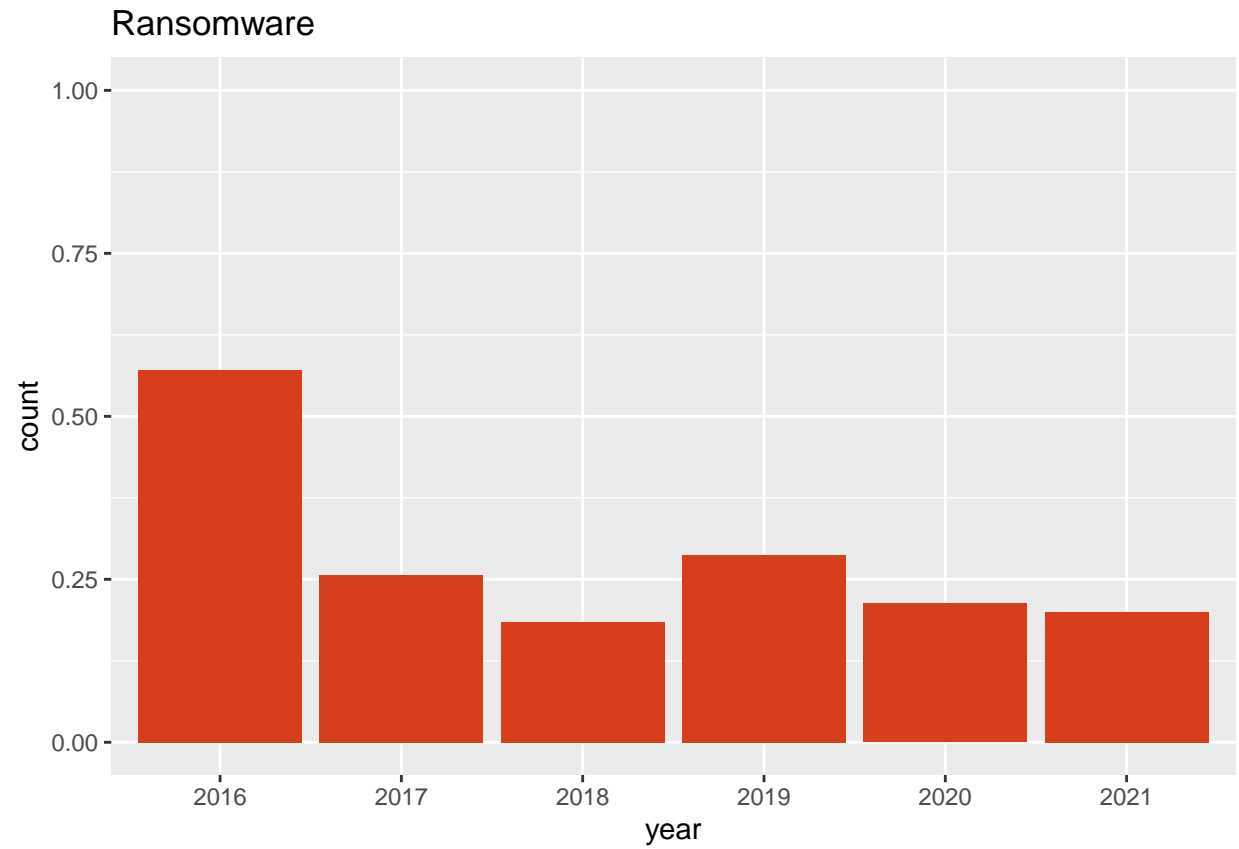
```
year.trend("Other", "#078c22")
```



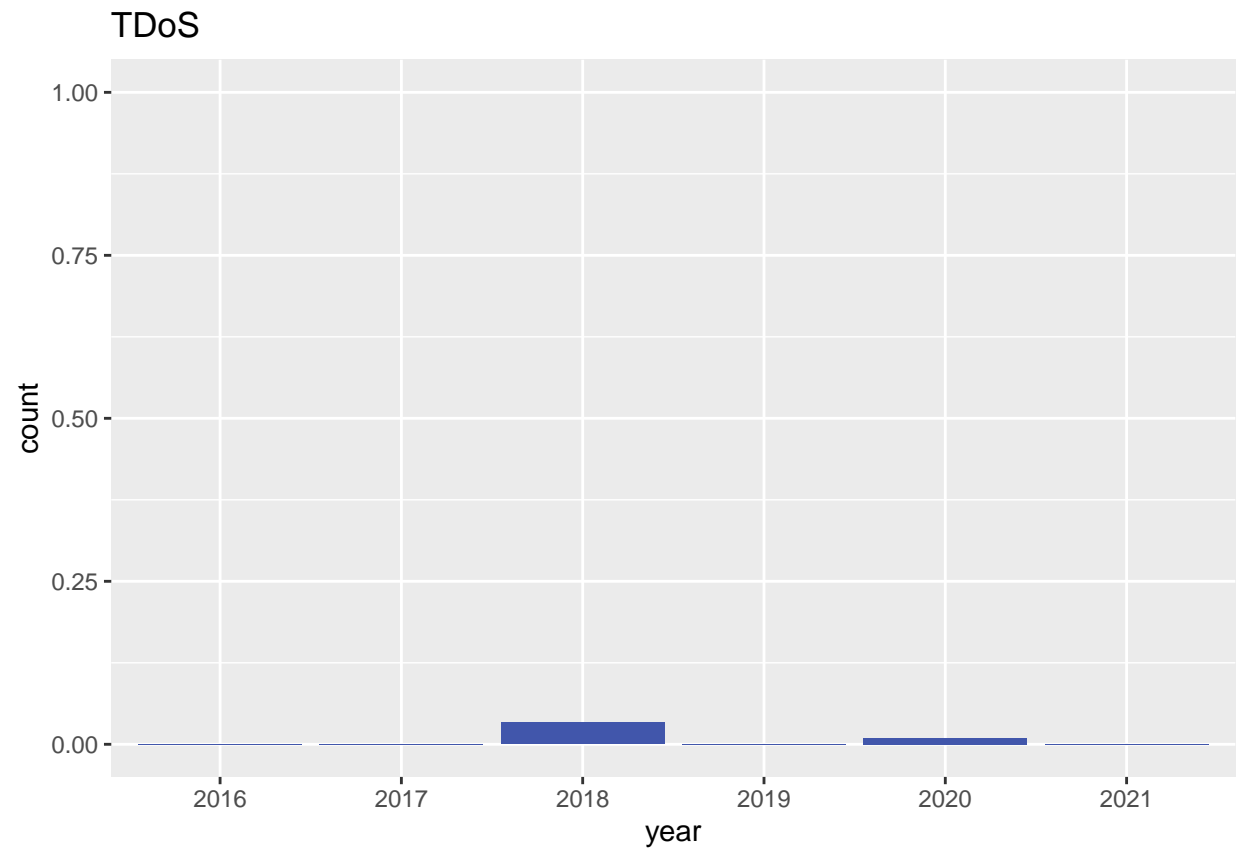
```
year.trend("Phishing", "#918511")
```



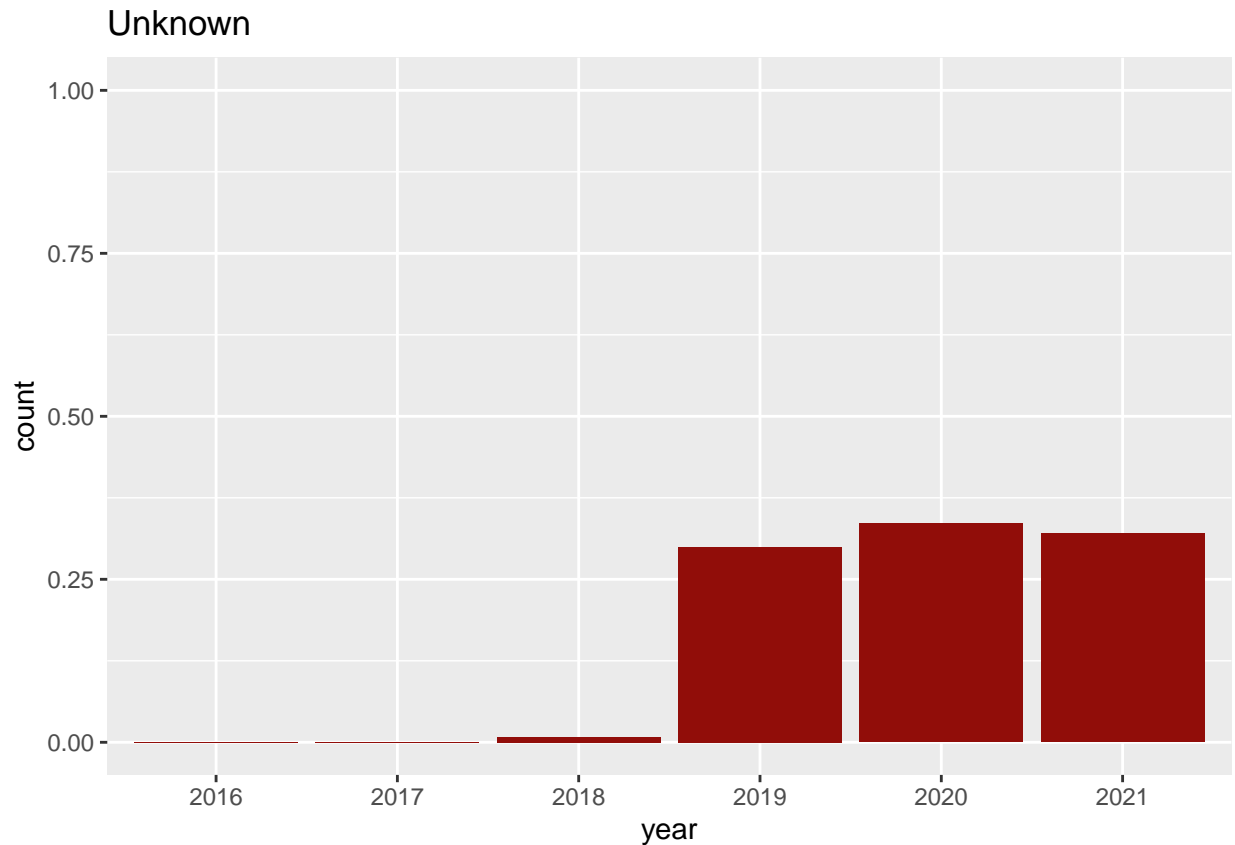
```
year.trend("Ransomware", "#d63e1c")
```



```
year.trend("TDoS", "#4156ab")
```



```
year.trend("Unknown", "#910d09")
```

```
#subset for education and one for safety
```

```
df.education <- subset(df.clean, type == "Education")
df.safety <- subset(df.clean, type == "Safety")
```

```
#a work around to not have to change a bunch of small code;
```

```
df.temporary <- df.clean
df.clean <- df.education
```

```
graph.attack.by.filter <- function(fill.choice){
  df.unk <- df.clean %>% as_tibble() %>%
  mutate(Unknown = as.factor(if_else(Unknown == 1, "True", "False")))
```

```
unk.graph <- ggplot(data = df.unk, aes(Unknown == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
```

```
df.hack <- df.clean %>% as_tibble() %>%
  mutate(Hack = as.factor(if_else(Hack == 1, "True", "False")))
```

```
hack.graph <- ggplot(data = df.hack, aes(Hack == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
```

```
df.data <- df.clean %>% as_tibble() %>%
  mutate(Data_Breach = as.factor(if_else(Data_Breach == 1, "True", "False")))
```

```
data.graph <- ggplot(data = df.data, aes(Data_Breach == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
```

```

df.ran <- df.clean %>% as_tibble() %>%
  mutate(Ransomware = as.factor(if_else(Ransomware == 1, "True", "False")))

ran.graph <- ggplot(data = df.ran, aes(Ransomware == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

df.phish <- df.clean %>% as_tibble() %>%
  mutate(Phishing = as.factor(if_else(Phishing == 1, "True", "False")))

phish.graph <- ggplot(data = df.phish, aes(Phishing == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

df.tdos <- df.clean %>% as_tibble() %>%
  mutate(TDoS = as.factor(if_else(TDoS == 1, "True", "False")))

tdos.graph <- ggplot(data = df.tdos, aes(TDoS == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

df.mal <- df.clean %>% as_tibble() %>%
  mutate(Malware = as.factor(if_else(Malware == 1, "True", "False")))

mal.graph <- ggplot(data = df.mal, aes(Malware == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

df.ddos <- df.clean %>% as_tibble() %>%
  mutate(DDoS = as.factor(if_else(DDoS == 1, "True", "False")))

ddos.graph <- ggplot(data = df.ddos, aes(DDoS == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

df.other <- df.clean %>% as_tibble() %>%
  mutate(Other = as.factor(if_else(Other == 1, "True", "False")))

other.graph <- ggplot(data = df.other, aes(Other == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

df.serv <- df.clean %>% as_tibble() %>%
  mutate(Server_Shutdown = as.factor(if_else(Server_Shutdown == 1, "True", "False")))

serv.graph <- ggplot(data = df.serv, aes(Server_Shutdown == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

(data.graph + scale_x_discrete(limit = c(TRUE)) | ddos.graph + scale_x_discrete(limit = c(TRUE)) |
  hack.graph + scale_x_discrete(limit = c(TRUE)))/ (mal.graph + scale_x_discrete(limit = c(TRUE)) |
  other.graph + scale_x_discrete(limit = c(TRUE)) | phish.graph + scale_x_discrete(limit = c(TRUE)))/
  (ran.graph + scale_x_discrete(limit = c(TRUE)) | tdos.graph + scale_x_discrete(limit = c(TRUE)) | unk
  scale_x_discrete(limit = c(TRUE)))
}

graph.attack.by.filter(df.clean$year)

```

```
## Warning: Removed 291 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 337 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 282 rows containing non-finite values (stat_count).
```

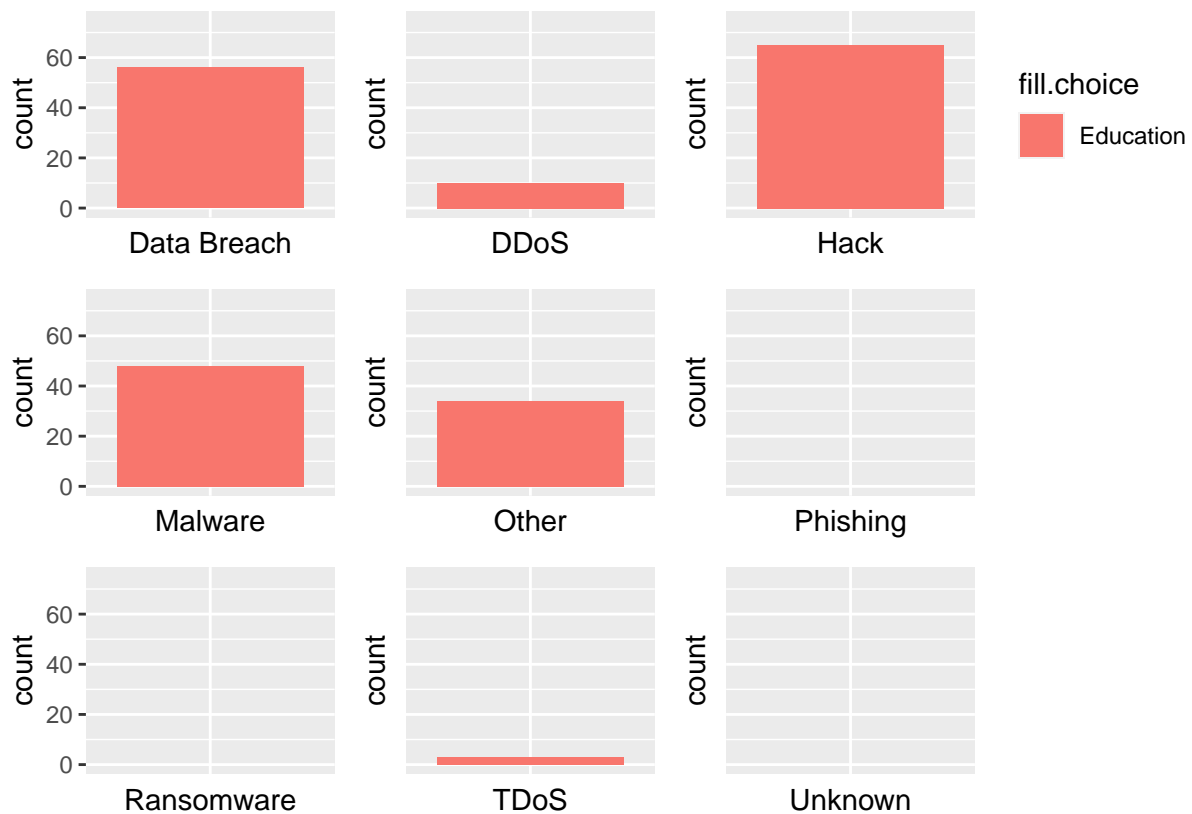
```
## Warning: Removed 299 rows containing non-finite values (stat_count).
## Warning: Removed 313 rows containing non-finite values (stat_count).
## Warning: Removed 248 rows containing non-finite values (stat_count).
## Warning: Removed 223 rows containing non-finite values (stat_count).
## Warning: Removed 344 rows containing non-finite values (stat_count).
## Warning: Removed 194 rows containing non-finite values (stat_count).
## Warning: Removed 1 rows containing missing values (geom_bar).
```



```
graph.attack.by.filter(df.clean$type)
```

```
## Warning: Removed 291 rows containing non-finite values (stat_count).
## Warning: Removed 337 rows containing non-finite values (stat_count).
## Warning: Removed 282 rows containing non-finite values (stat_count).
## Warning: Removed 299 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 313 rows containing non-finite values (stat_count).
## Warning: Removed 248 rows containing non-finite values (stat_count).
## Warning: Removed 1 rows containing missing values (geom_bar).
## Warning: Removed 223 rows containing non-finite values (stat_count).
## Warning: Removed 1 rows containing missing values (geom_bar).
## Warning: Removed 344 rows containing non-finite values (stat_count).
## Warning: Removed 194 rows containing non-finite values (stat_count).
## Warning: Removed 1 rows containing missing values (geom_bar).
```



```
graph.attack.by.filter((df.clean$website_quality))
```

```
## Warning: Removed 291 rows containing non-finite values (stat_count).
## Warning: Removed 337 rows containing non-finite values (stat_count).
## Warning: Removed 282 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 299 rows containing non-finite values (stat_count).

## Warning: Removed 313 rows containing non-finite values (stat_count).

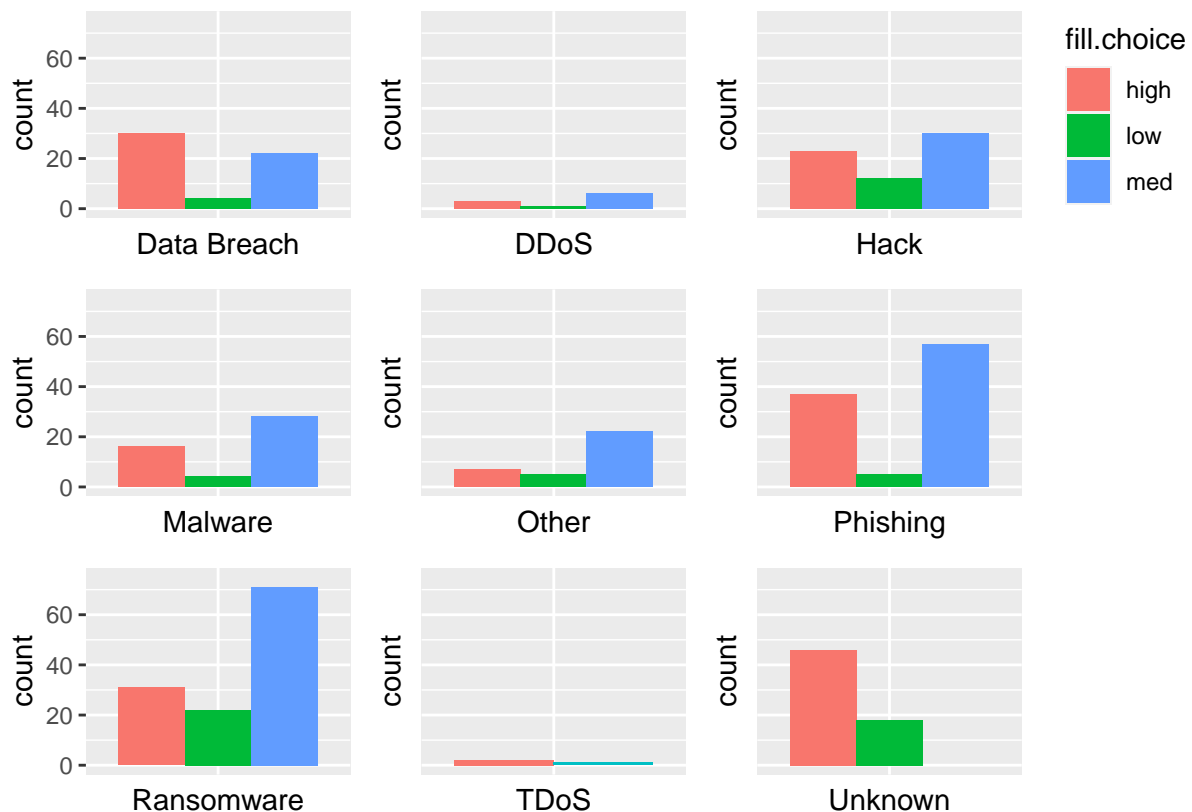
## Warning: Removed 248 rows containing non-finite values (stat_count).

## Warning: Removed 223 rows containing non-finite values (stat_count).

## Warning: Removed 344 rows containing non-finite values (stat_count).

## Warning: Removed 194 rows containing non-finite values (stat_count).

## Warning: Removed 1 rows containing missing values (geom_bar).
```



```
df.clean <- df.temporary
```

```
#a work around to not have to change a bunch of small code;
```

```
df.temporary <- df.clean
df.clean <- df.safety
```

```
graph.attack.by.filter <- function(fill.choice){
  df.unk <- df.clean %>% as_tibble() %>%
```

```

mutate(Unknown = as.factor(if_else(Unknown == 1, "True", "False")))

unk.graph <- ggplot(data = df.unk, aes(Unknown == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.hack <- df.clean %>% as_tibble() %>%
  mutate(Hack = as.factor(if_else(Hack == 1, "True", "False")))

hack.graph <- ggplot(data = df.hack, aes(Hack == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.data <- df.clean %>% as_tibble() %>%
  mutate(Data_Breach = as.factor(if_else(Data_Breach == 1, "True", "False")))

data.graph <- ggplot(data = df.data, aes(Data_Breach == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.ran <- df.clean %>% as_tibble() %>%
  mutate(Ransomware = as.factor(if_else(Ransomware == 1, "True", "False")))

ran.graph <- ggplot(data = df.ran, aes(Ransomware == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.phish <- df.clean %>% as_tibble() %>%
  mutate(Phishing = as.factor(if_else(Phishing == 1, "True", "False")))

phish.graph <- ggplot(data = df.phish, aes(Phishing == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.tdos <- df.clean %>% as_tibble() %>%
  mutate(TDoS = as.factor(if_else(TDoS == 1, "True", "False")))

tdos.graph <- ggplot(data = df.tdos, aes(TDoS == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.mal <- df.clean %>% as_tibble() %>%
  mutate(Malware = as.factor(if_else(Malware == 1, "True", "False")))

mal.graph <- ggplot(data = df.mal, aes(Malware == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.ddos <- df.clean %>% as_tibble() %>%
  mutate(DDoS = as.factor(if_else(DDoS == 1, "True", "False")))

ddos.graph <- ggplot(data = df.ddos, aes(DDoS == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.other <- df.clean %>% as_tibble() %>%
  mutate(Other = as.factor(if_else(Other == 1, "True", "False")))

other.graph <- ggplot(data = df.other, aes(Other == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####
df.serv <- df.clean %>% as_tibble() %>%
  mutate(Server_Shutdown = as.factor(if_else(Server_Shutdown == 1, "True", "False")))

serv.graph <- ggplot(data = df.serv, aes(Server_Shutdown == "True", fill = fill.choice))+geom_bar(position = "dodge")
#####

(data.graph + scale_x_discrete(limit = c(TRUE)) | ddos.graph + scale_x_discrete(limit = c(TRUE)) |
  hack.graph + scale_x_discrete(limit = c(TRUE)))/ (mal.graph + scale_x_discrete(limit = c(TRUE)) |

```

```

other.graph + scale_x_discrete(limit = c(TRUE)) | phish.graph + scale_x_discrete(limit = c(TRUE))/
(ran.graph + scale_x_discrete(limit = c(TRUE)) | tdos.graph + scale_x_discrete(limit = c(TRUE)) | unk
scale_x_discrete(limit = c(TRUE)))
}

graph.attack.by.filter(df.clean$year)

```

```

## Warning: Removed 165 rows containing non-finite values (stat_count).

## Warning: Removed 192 rows containing non-finite values (stat_count).

## Warning: Removed 132 rows containing non-finite values (stat_count).

## Warning: Removed 167 rows containing non-finite values (stat_count).

## Warning: Removed 172 rows containing non-finite values (stat_count).

## Warning: Removed 184 rows containing non-finite values (stat_count).

## Warning: Removed 94 rows containing non-finite values (stat_count).

## Warning: Removed 190 rows containing non-finite values (stat_count).

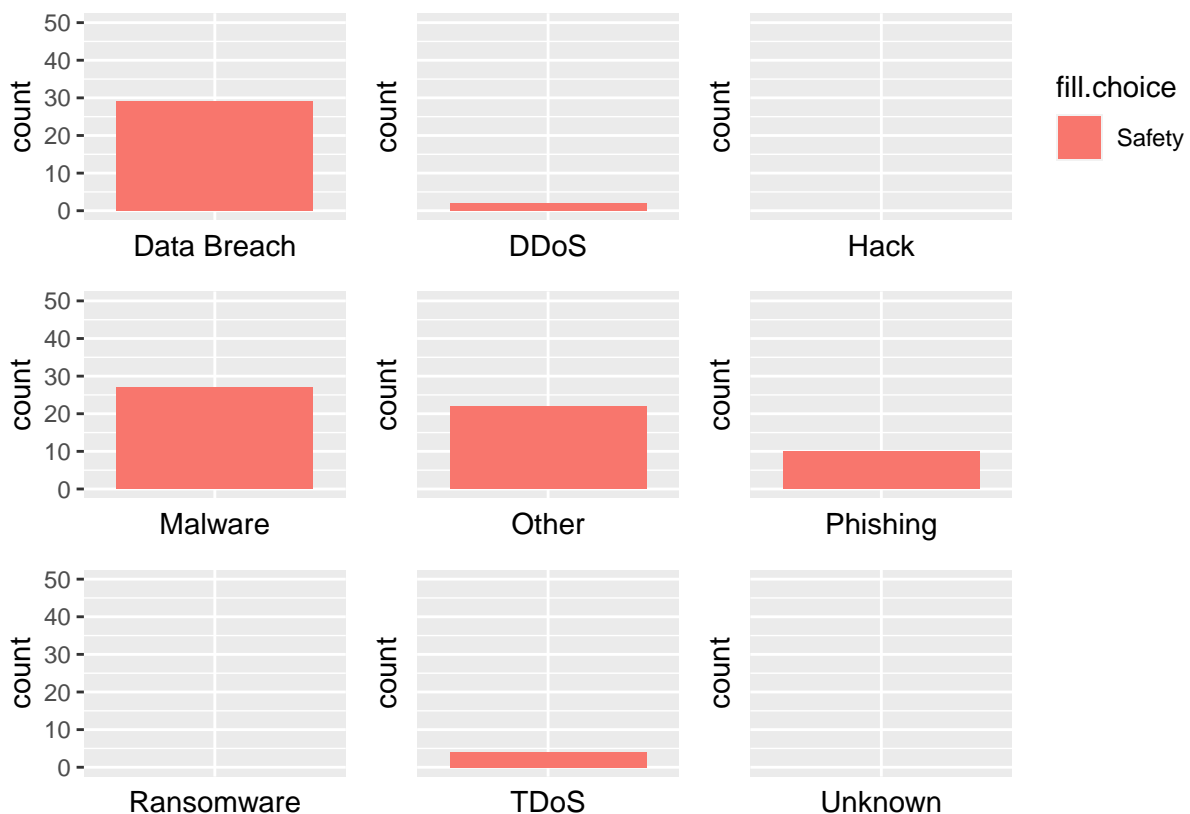
## Warning: Removed 126 rows containing non-finite values (stat_count).

```



```
graph.attack.by.filter(df.clean$type)
```

```
## Warning: Removed 165 rows containing non-finite values (stat_count).  
## Warning: Removed 192 rows containing non-finite values (stat_count).  
## Warning: Removed 132 rows containing non-finite values (stat_count).  
## Warning: Removed 1 rows containing missing values (geom_bar).  
## Warning: Removed 167 rows containing non-finite values (stat_count).  
## Warning: Removed 172 rows containing non-finite values (stat_count).  
## Warning: Removed 184 rows containing non-finite values (stat_count).  
## Warning: Removed 94 rows containing non-finite values (stat_count).  
## Warning: Removed 1 rows containing missing values (geom_bar).  
## Warning: Removed 190 rows containing non-finite values (stat_count).  
## Warning: Removed 126 rows containing non-finite values (stat_count).  
## Warning: Removed 1 rows containing missing values (geom_bar).
```




```
graph.attack.by.filter((df.clean$website_quality))
```

```
## Warning: Removed 165 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 192 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 132 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 167 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 172 rows containing non-finite values (stat_count).
```

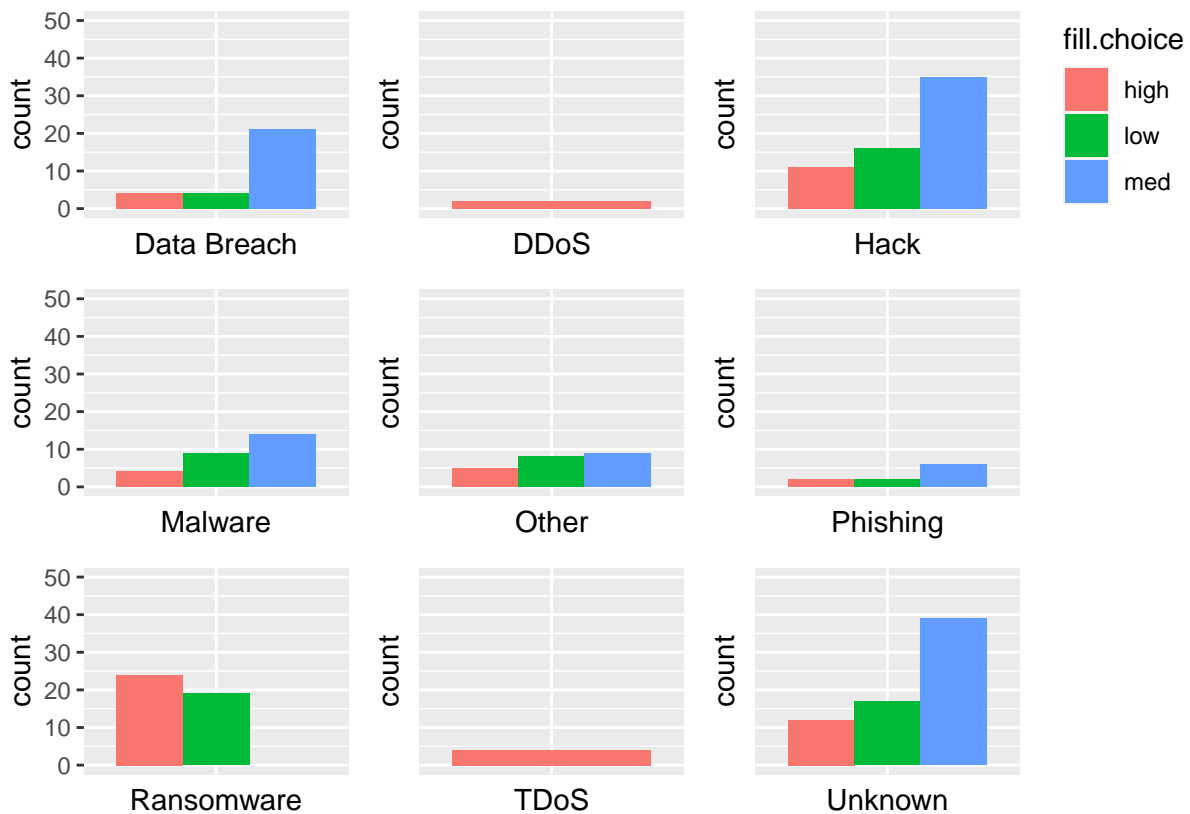
```
## Warning: Removed 184 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 94 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 1 rows containing missing values (geom_bar).
```

```
## Warning: Removed 190 rows containing non-finite values (stat_count).
```

```
## Warning: Removed 126 rows containing non-finite values (stat_count).
```



```
df.clean <- df.temporary
```

```
get.percent <- function(attack.type, sector){  
  
  sum.type <- sum(df.clean$type == sector)  
  
  sum.attack <- sum(df.clean[which(df.clean$type == sector), attack.type])  
  percent.attack = sum.attack/sum.type;  
  
  #sprintf("%f of % s attacks are % s", percent.attack, sector, attack.type )  
  return(percent.attack)  
}
```

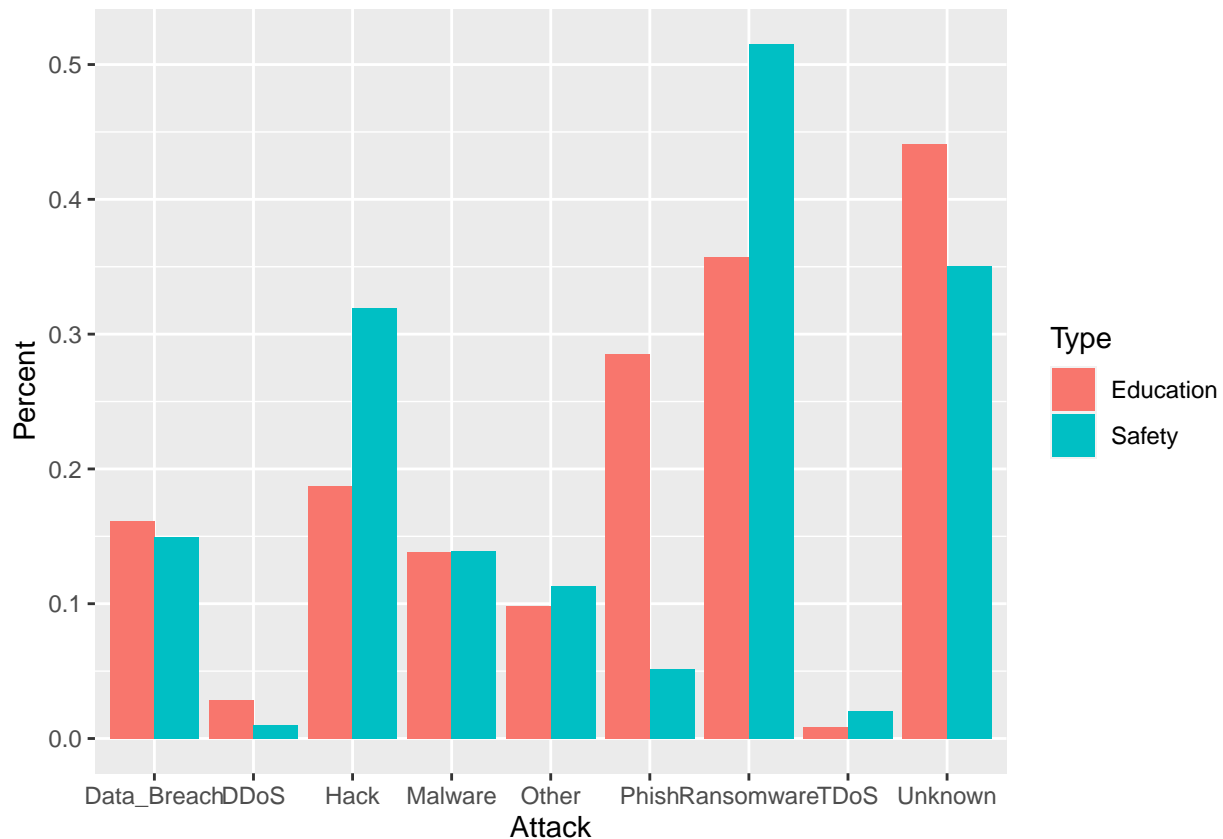
```
df.attack.count <- data.frame(matrix(ncol = 3, nrow = 18))  
x <- c("Attack", "Type", "Percent")  
colnames(df.attack.count)<- x  
  
df.attack.count[1, "Attack"] = "Data_Breach"  
df.attack.count[2, "Attack"] = "DDoS"  
df.attack.count[3, "Attack"] = "Hack"  
df.attack.count[4, "Attack"] = "Malware"  
df.attack.count[5, "Attack"] = "Other"  
df.attack.count[6, "Attack"] = "Phish"  
df.attack.count[7, "Attack"] = "Ransomware"  
df.attack.count[8, "Attack"] = "TDoS"  
df.attack.count[9, "Attack"] = "Unknown"  
df.attack.count[10, "Attack"] = "Data_Breach"  
df.attack.count[11, "Attack"] = "DDoS"  
df.attack.count[12, "Attack"] = "Hack"  
df.attack.count[13, "Attack"] = "Malware"  
df.attack.count[14, "Attack"] = "Other"  
df.attack.count[15, "Attack"] = "Phish"  
df.attack.count[16, "Attack"] = "Ransomware"  
df.attack.count[17, "Attack"] = "TDoS"  
df.attack.count[18, "Attack"] = "Unknown"  
df.attack.count[1:9, "Type"] = "Education"  
df.attack.count[10:18, "Type"] = "Safety"  
  
df.attack.count[1, "Percent"] = get.percent("Data_Breach", "Education")  
df.attack.count[2, "Percent"] = get.percent("DDoS", "Education")  
df.attack.count[3, "Percent"] = get.percent("Hack", "Education")  
df.attack.count[4, "Percent"] = get.percent("Malware", "Education")  
df.attack.count[5, "Percent"] = get.percent("Other", "Education")  
df.attack.count[6, "Percent"] = get.percent("Phishing", "Education")  
df.attack.count[7, "Percent"] = get.percent("Ransomware", "Education")  
df.attack.count[8, "Percent"] = get.percent("TDoS", "Education")  
df.attack.count[9, "Percent"] = get.percent("Unknown", "Education")  
  
df.attack.count[10, "Percent"] = get.percent("Data_Breach", "Safety")  
df.attack.count[11, "Percent"] = get.percent("DDoS", "Safety")  
df.attack.count[12, "Percent"] = get.percent("Hack", "Safety")  
df.attack.count[13, "Percent"] = get.percent("Malware", "Safety")
```

```
df.attack.count[14, "Percent"] = get.percent("Other", "Safety")
df.attack.count[15, "Percent"] = get.percent("Phishing", "Safety")
df.attack.count[16, "Percent"] = get.percent("Ransomware", "Safety")
df.attack.count[17, "Percent"] = get.percent("TDoS", "Safety")
df.attack.count[18, "Percent"] = get.percent("Unknown", "Safety")
```

```
head(df.attack.count)
```

```
##      Attack      Type      Percent
## 1 Data_Breach Education 0.16138329
## 2      DDoS Education 0.02881844
## 3      Hack Education 0.18731988
## 4    Malware Education 0.13832853
## 5      Other Education 0.09798271
## 6      Phish Education 0.28530259
```

```
ggplot(df.attack.count, aes(x = Attack, y = Percent, fill = Type)) + geom_bar(stat = "identity", position = "dodge")
```



```
get.percent <- function(quality, sector){
  sum.type <- sum(df.clean$type == sector)
  sum.attack <- sum(df.clean[which(df.clean$type == sector), quality])
  percent.attack = sum.attack/sum.type;
}
```

```
#sprintf("%f of % s attacks are % s", percent.attack, sector, attack.type )
return(percent.attack)
}
```

```
df.quality.count <- data.frame(matrix(ncol = 3, nrow = 6))
x <- c("Type", "Quality", "Percent")
colnames(df.quality.count)<- x

df.quality.count[1, "Quality"] = "Low"
df.quality.count[2, "Quality"] = "Medium"
df.quality.count[3, "Quality"] = "High"
df.quality.count[4, "Quality"] = "Low"
df.quality.count[5, "Quality"] = "Medium"
df.quality.count[6, "Quality"] = "High"
df.quality.count[1:3, "Type"] = "Education"
df.quality.count[4:6, "Type"] = "Safety"

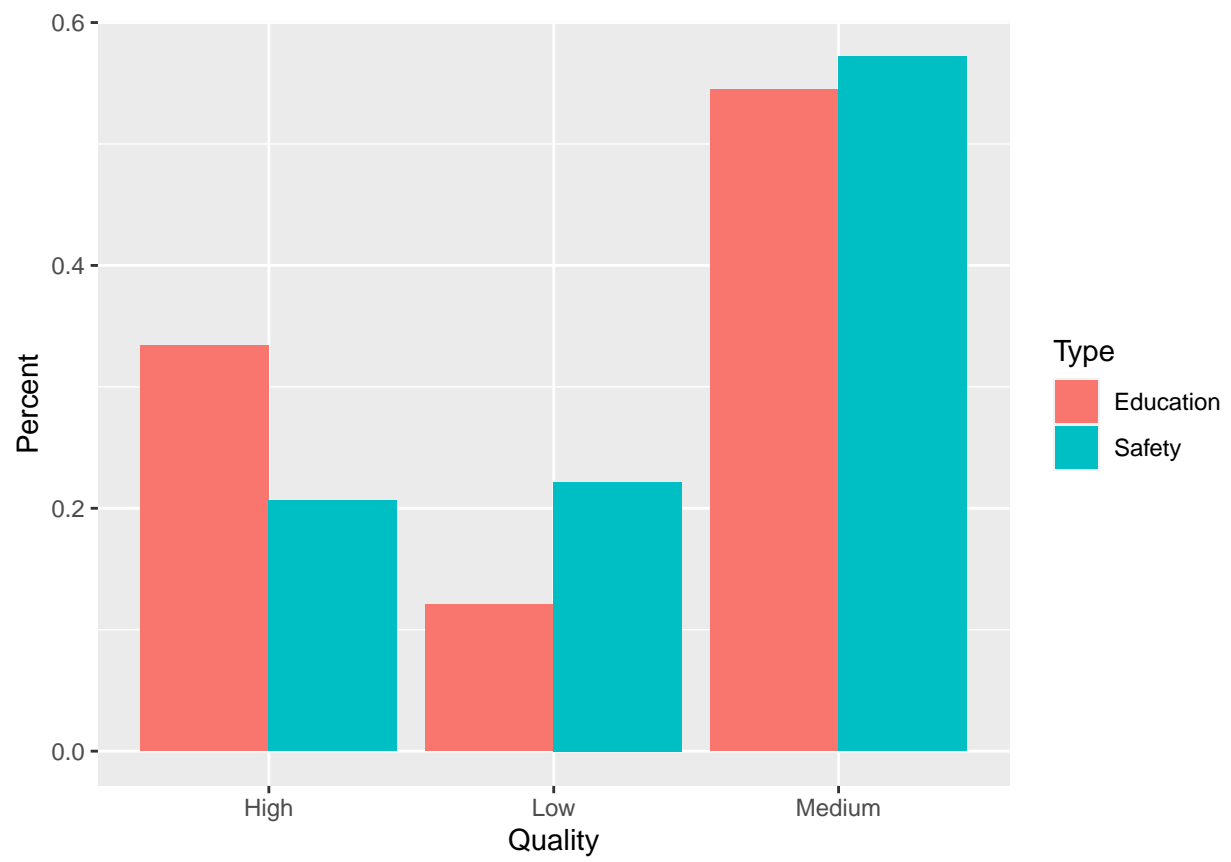
df.quality.count[1, "Percent"] = get.percent("low", "Education")
df.quality.count[2, "Percent"] = get.percent("med", "Education")
df.quality.count[3, "Percent"] = get.percent("high", "Education")

df.quality.count[4, "Percent"] = get.percent("low", "Safety")
df.quality.count[5, "Percent"] = get.percent("med", "Safety")
df.quality.count[6, "Percent"] = get.percent("high", "Safety")
```

```
head(df.quality.count)
```

```
##      Type Quality  Percent
## 1 Education    Low 0.1210375
## 2 Education  Medium 0.5446686
## 3 Education    High 0.3342939
## 4   Safety     Low 0.2216495
## 5   Safety  Medium 0.5721649
## 6   Safety     High 0.2061856
```

```
ggplot(df.quality.count, aes(x = Quality, y = Percent, fill = Type)) + geom_bar(stat = "identity", posi
```



““