

# Projet Sécurité des sites de Villes

Dahiez Ulysse , Burdy Simon

02/12/2020

## Introduction

Nous avons choisi de travailler sur les données des niveaux de sécurité (présences https , versions langages , serveurs à jour , etc ...) des sites de Mairie de toutes les villes et villages de France. Ces données proviennent du site gouvernemental "<https://www.data.gouv.fr/fr/>".

Nous nous sommes posés les questions suivantes :

La taille de la population par villes/villages a-t-elle une influence sur les niveaux de sécurité de leurs sites de Mairies ?

La position géographique a-t-elle une influence sur les niveaux de sécurité de leurs sites de Mairies ?

```
villeSecuInfo <- read_csv(file = "ville_secu_informatique.csv")
```

```
## Parsed with column specification:
## cols(
##   Commune = col_character(),
##   'Code Insee' = col_character(),
##   url = col_character(),
##   Catégorie = col_character(),
##   Population = col_double(),
##   Site = col_character(),
##   https = col_character(),
##   Serveur = col_character(),
##   'Version du serveur' = col_character(),
##   Application = col_character(),
##   'Version de l'application' = col_character(),
##   Langage = col_character(),
##   'Version du langage' = col_character(),
##   Latitude = col_double(),
##   Longitude = col_double()
## )
```

```
villeSecuInfohttpsnon <- filter(villeSecuInfo, Population > 25000 & https == "non" )
villeSecuInfohttpsoui <- filter(villeSecuInfo, Population > 25000 & https == "oui" )
```

```
map(database="france" )
symbols(villeSecuInfohttpsnon$Longitude,
        villeSecuInfohttpsnon$Latitude,
        bg="#e2373f",
        fg="#ffffff",
```

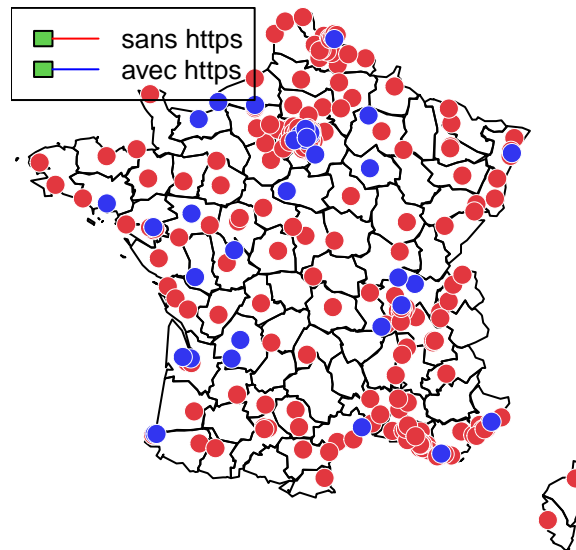
```

lwd=0.5,
circles=rep(0.2,
            length(villeSecuInfohttpsnon$Longitude)),
inches=0.05, add=TRUE)

symbols(villeSecuInfohttpsoui$Longitude,
        villeSecuInfohttpsoui$Latitude,
        bg="#3235f0",
        fg="ffffff",
        lwd=0.5,
        circles=rep(0.2,
                    length(villeSecuInfohttpsoui$Longitude)),
        inches=0.05, add=TRUE)
legend("topleft", 5, 51, legend=c("sans https", "avec https"),
      col=c("red", "blue"), lty=1:1, cex=0.8)
title(main = "Position des villes de plus de 25 000 habitants \nayant une adresse https")

```

## Position des villes de plus de 25 000 habitants ayant une adresse https



D'après cette carte nous observons une majorité de sites n'ayant pas de HTTPS . Nous allons ensuite étudier cette remarque plus en détail dans les parties suivantes .

# Influence de la taille de la Population sur le niveau de sécurité globale :

## Langage utilisé

### 1. Ville de moins de 5000 habitants :

Tout d'abors j'ai voulu observer les différents langages utilisés par les sites web de villes de moins de 5000 habitant en fonction de leur catégories de sécurité .

```
# Filter pour a retirer les données inconnues
villeLangage <- filter(villeSecuInfo , Langage != "Inconnue" , Catégorie != "Inconnue")
petiteVilleLangage <- filter(villeLangage , Population <= 5000)

#connaître le pourcentage de langage utilisées
count <- table(unlist(petiteVilleLangage$Langage))
perc <- 100*count/sum(count)
result <- data.frame(code = sprintf('%10s', names(count)),
                     count = as.integer(count), perc = as.numeric(perc))

attach(result)
```

```
## The following objects are masked _by_ .GlobalEnv:
##
##      count, perc
```

```
prcLangage <- result[order(-perc),]

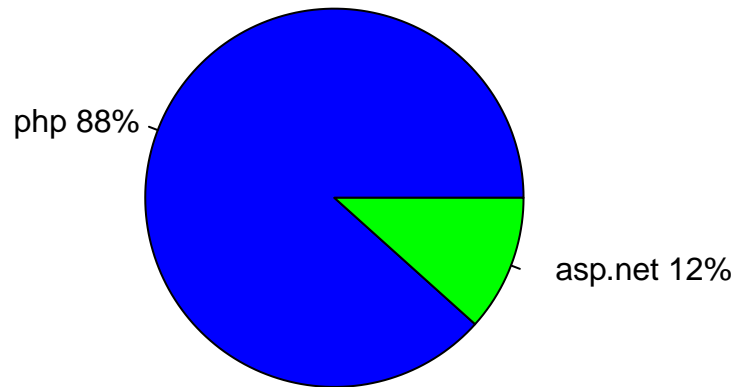
detach(result)

prcLangage <- prcLangage[1:2,]

### Camembert % langage
slices <- c(prcLangage$perc)
lbls <- c(prcLangage$code)
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels

pie(prcLangage$perc , col =c("blue","green"),
    labels = lbls ,main=paste("Proportion des langages utilisés \npour " ,
nrow(petiteVilleLangage) ," Villes de moins de 5000 habitants " ) ,xpd = TRUE)
```

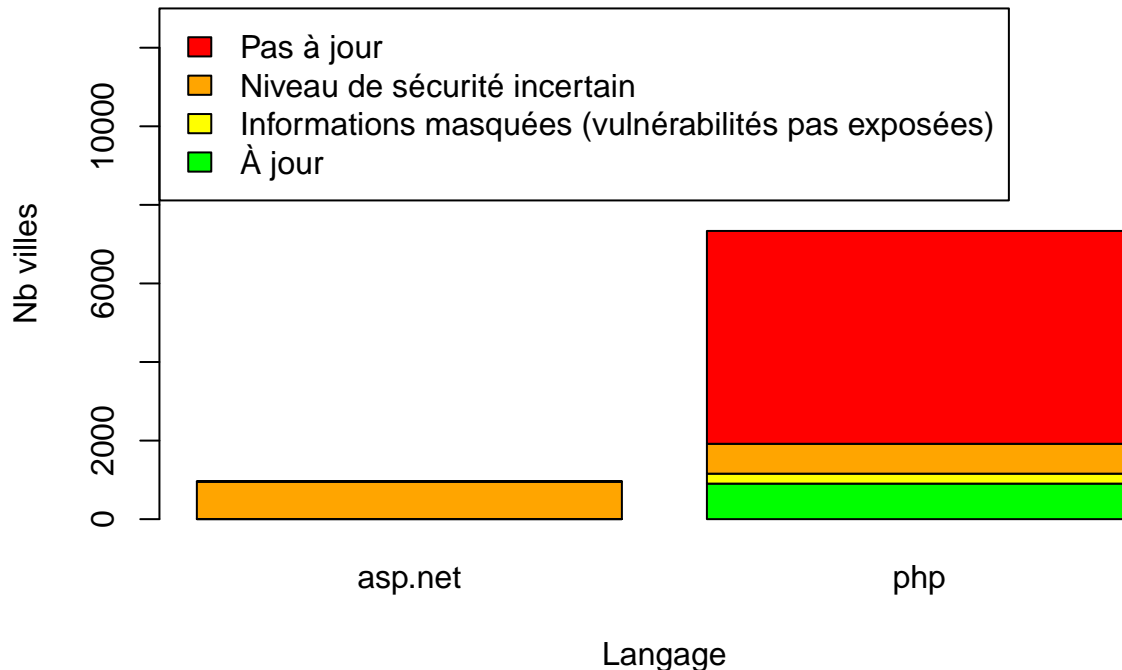
## Proportion des langages utilisés pour 8303 Villes de moins de 5000 habitants



Plus de 80% des villes de moins de 5000 habitants utilisent le langage PHP et seulement 12 % le langage asp.net. Nous allons maintenant observer les proportions des différentes catégories de sécurité pour chaque langage.

```
barplot(table(petiteVilleLangage$Catégorie,petiteVilleLangage$Langage) ,
col =c("green","yellow","orange","red") ,
legend.text = c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ),
args.legend = list(x = "topleft") ,
main=paste( "Graphique montrant les catégories de degrés de mise à \njours des sites de" ,
nrow(petiteVilleLangage) ," Villes de moins de 5000 habitants\n en fonction de leur langage" ),
xlab = "Langage" ,
ylab="Nb villes" , ylim=c(0,13000))
```

## Graphique montrant les catégories de degrés de mise à jour des sites de 8303 Villes de moins de 5000 habitants en fonction de leur langage



On observe qu'une très grande majorité des sites utilisent le langage PHP mais qu'elles restent loin du niveau de sécurité maximum (À jour). On observe aussi que la minorité des sites utilisant le langage asp.net sont tous "d'un niveau de sécurité incertain". On se rends bien compte que très peu de villes (moins de 1000 villes) ont un site "À jour".

Pour mieux observer les proportions sur le langage PHP :

```
petiteVilleLangCatPhp <- filter(petiteVilleLangage , Langage == "php")

count <- table(unlist(petiteVilleLangCatPhp$Catégorie))
perc <- 100*count/sum(count)
result <- data.frame(code = sprintf('%10s', names(count)),
                     count = as.integer(count), perc = as.numeric(perc))

attach(result)
```

```
## The following objects are masked _by_ .GlobalEnv:
##
## count, perc
```

```
prcCat <- result[order(-perc),]

detach(result)

prcCat <- prcCat[1:4,]
```

```

slices <- c(prcCat$perc)
lbls <- c(prcCat$code)
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels

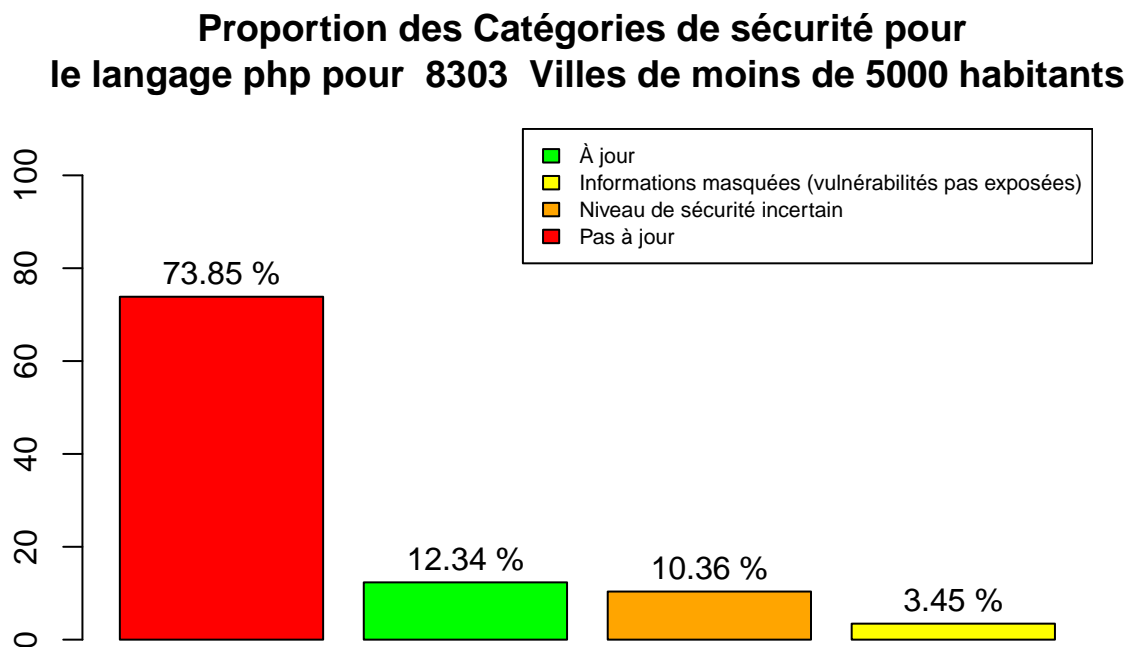
phpLang <- barplot(prcCat$perc , col =c("red" ,"green","orange","yellow"),
main=paste("Proportion des Catégories de sécurité pour \n le langage php pour " ,
nrow(petiteVilleLangage) ," Villes de moins de 5000 habitants " ) ,
ylim=c(0,110) )

percValeurs<-as.matrix(prcCat$perc)

text(phpLang,percValeurs+5,labels=as.character(paste(round(percValeurs,2),"%")) )

legend( "topright" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) ,
fill = c("green","yellow","orange","red") ,cex=0.70)

```



Pour le langage asp.net :

```

petiteVilleLangCatAsp <- filter(petiteVilleLangage , Langage == "asp.net")
count <- table(unlist(petiteVilleLangCatAsp$Catégorie))

```

```
perc <- 100*count/sum(count)
result <- data.frame(code = sprintf('%10s', names(count)),
                     count = as.integer(count), perc = as.numeric(perc))
```

```
attach(result)
```

```
## The following objects are masked _by_ .GlobalEnv:
##
##      count, perc
```

```
prcCat <- result[order(-perc),]
```

```
detach(result)
```

```
prcCat <- prcCat[1:4,]
```

```
slices <- c(prcCat$perc)
lbls <- c(prcCat$code)
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls, "%", sep=" ") # ad % to labels
```

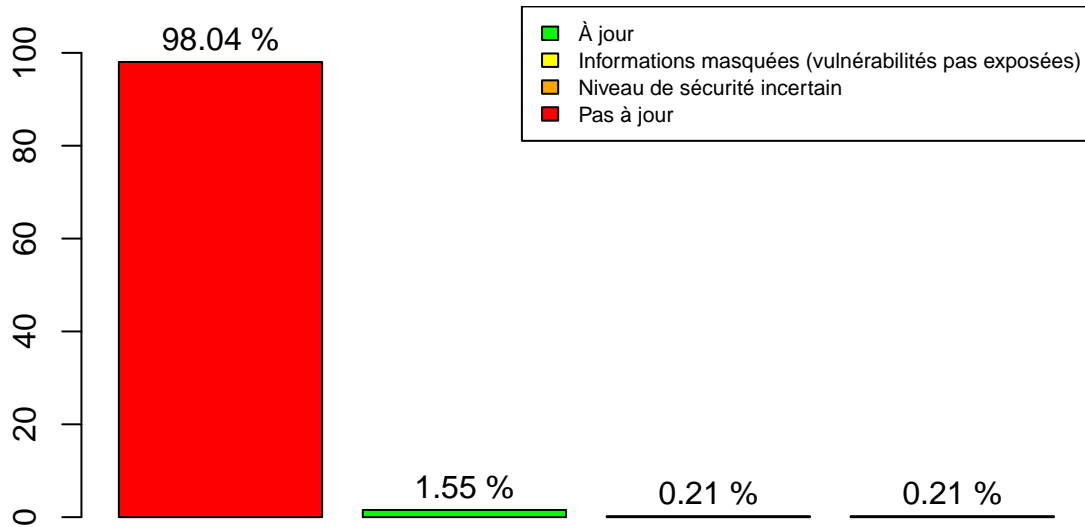
```
aspLang <- barplot(prcCat$perc , col =c("red" ,"green","orange","yellow"),
main=paste("Proportion des Catégories de sécurité pour \n le langage asp.net pour "
, nrow(petiteVilleLangage) ," Villes de moins de 5000 habitants " ),
ylim=c(0,110) , cex.names = 0.5)
```

```
percValeurs<-as.matrix(prcCat$perc)
```

```
text(aspLang,percValeurs+5,labels=as.character(paste(round(percValeurs,2),"%")) )
```

```
legend( "topright" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) ,
fill = c("green","yellow","orange","red") ,cex=0.70)
```

## Proportion des Catégories de sécurité pour le langage asp.net pour 8303 Villes de moins de 5000 habitants



### 2. Ville de plus de 150 000 habitants :

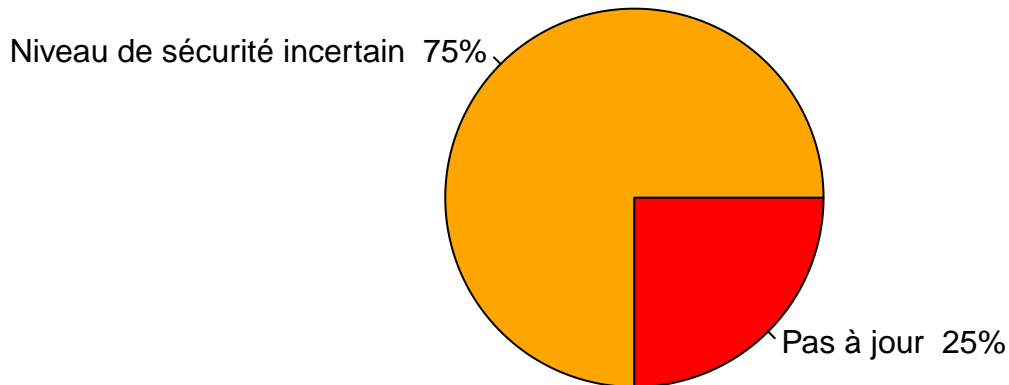
Nous regardons maintenant pour les villes de plus de 150 000 habitants .

```
villeLangage <- filter(villeSecuInfo , Langage != "Inconnue" , Catégorie != "Inconnue")
grandeVilleLangage <- filter(villeLangage , Population > 150000)
slices <- c(75,25)
lbls <- c("Niveau de sécurité incertain ", "Pas à jour ")
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels

pie(table(grandeVilleLangage$Catégorie,grandeVilleLangage$Langage) , col =c("orange","red"),
labels = lbls ,
main=paste("Graphique Circulaire montrant les catégories de degrés de mise à \njours des sites de ",
nrow(grandeVilleLangage) ," Villes de plus de 150 000 habitants \nen fonction de leur langage" ))
```



## Graphique Circulaire montrant les catégories de degrés de mise à jour des sites de 4 Villes de plus de 150 000 habitants en fonction de leur langage



On peut déjà remarquer que seulement 4 villes de plus de 150 000 habitants correspondent aux critères demandés qui sont d'avoir une "Catégorie de sécurité" et un "langage" utilisés différents de "Inconnue". La totalité de ces villes utilise le langage PHP, malgré le fait d'être des sites web de grandes villes aucun de ces sites web n'est dans la catégorie "À jour".

### Serveur

Maintenant on va observer les différents serveurs en fonction de la catégorie de sécurité.

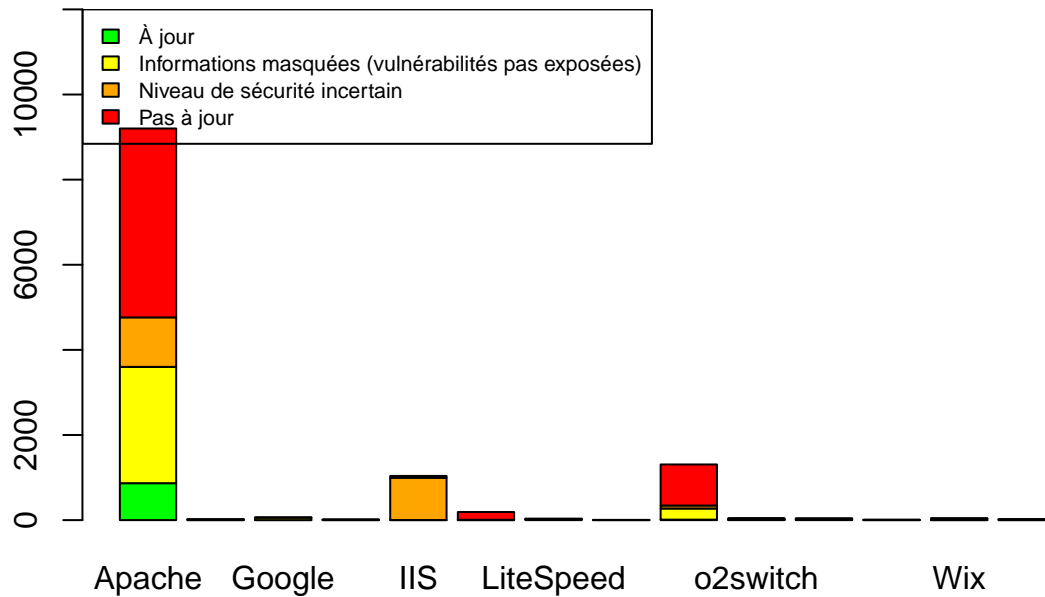
#### 1. Villes de moins de 5000 habitants :

On va observer les serveurs utilisés pour les villes de moins de 5000 habitants.

```
villeServeur <- filter(villeSecuInfo, Serveur != "Inconnue", Catégorie != "Inconnue")
petiteVilleServeur <- filter(villeServeur, Population < 5000)
barplot(table(petiteVilleServeur$Catégorie, petiteVilleServeur$Serveur),
col = c("green", "yellow", "orange", "red"),
main = paste("Niveau de mise à jours des", nrow(petiteVilleServeur),
" serveurs des sites de ville"),
ylim = c(0, 12000),
)
legend("topleft", c('À jour', "Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain", "Pas à jour"),
```

```
fill = c("green", "yellow", "orange", "red") ,
cex=0.70)
```

## Niveau de mise à jours des 12016 serveurs des sites de ville



On remarque que la majorité des villes utilisent le serveur Apache , qui est en très grande partie composé de sites “Pas à jour” . On remarque tout de même une proportion de sites où les informations sont masquées.

On regarde plus précisément pour Apache :

```
petiteVilleServeur <- filter( villeServeur, Population < 5000 , Serveur == "Apache" )

count <- table(unlist(petiteVilleServeur$Catégorie))
perc <- 100*count/sum(count)
result <- data.frame(code = sprintf('%10s', names(count)),
                     count = as.integer(count), perc = as.numeric(perc))

attach(result)
```

```
## The following objects are masked _by_ .GlobalEnv:
##
## count, perc
```

```
prcCat <- result[order(-perc),]
```

```
detach(result)

prcCat <-prcCat[1:4,]

slices <-c(prcCat$perc)
lbls <- c(prcCat$code)
pct <- pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels

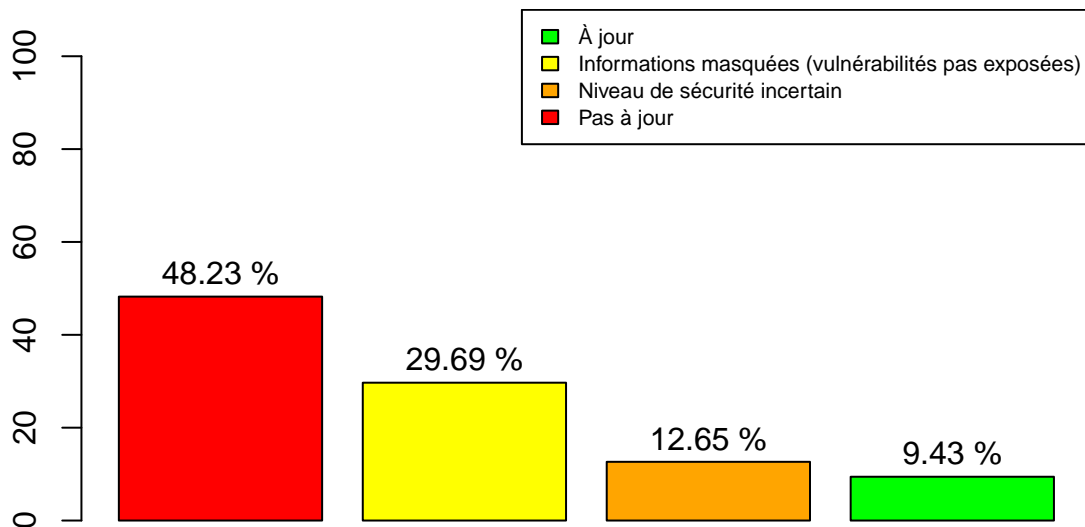
apacheCat <- barplot(prcCat$perc , col =c("red" ,"yellow","orange","green"),
main=paste("Proportion des Catégories de sécurité pour \n le serveur Apache pour " , nrow(petiteVilleSe

percValeurs<-as.matrix(prcCat$perc)

text(apacheCat ,percValeurs+5,labels=as.character(paste(round(percValeurs,2),"%")) )

legend( "topright" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) , fill = c("green","yellow","orange","red") ,cex=0.70
```

### Proportion des Catégories de sécurité pour le serveur Apache pour 9202 Villes de moins de 5000 habitants

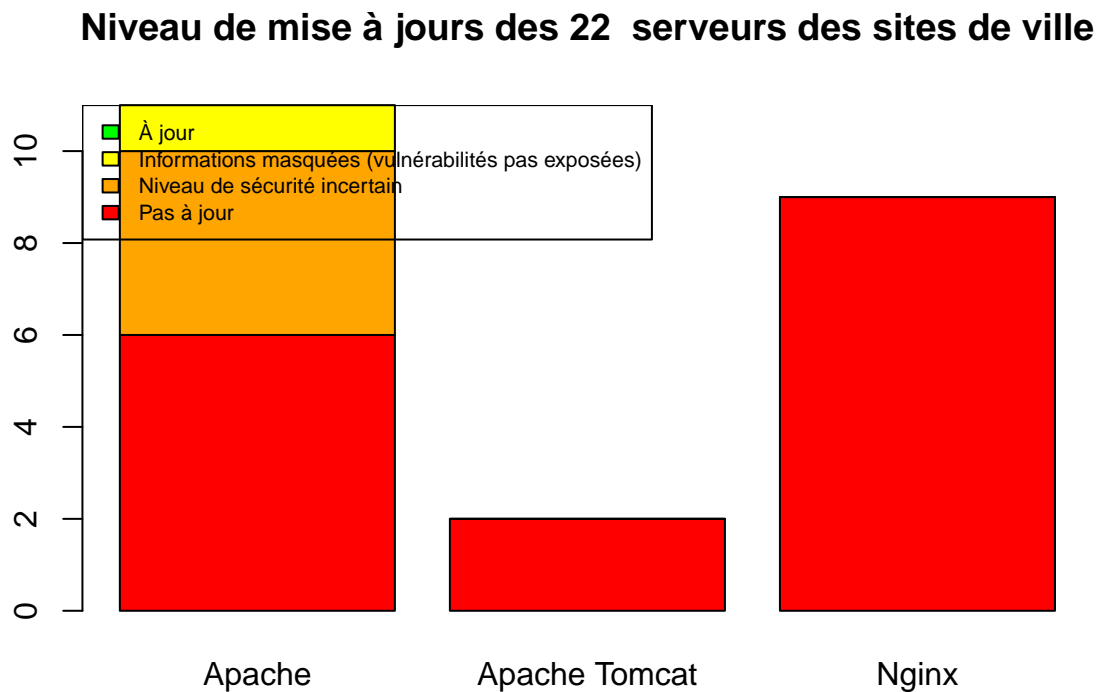


Comme dit précédemment plus de 48 % sont classé comme " Pas à Jour" pour seulement de 9.43% de "À jour" .

## 2. Villes de plus de 150 000 habitants :

On vas observer les serveurs utilisés pour les villes de plus de 150 000 habitants .

```
villeServeur <- filter(villeSecuInfo , Serveur != "Inconnue" , Catégorie != "Inconnue")
grandeVilleServeur <- filter(villeServeur , Population > 150000 )
barplot(table( grandeVilleServeur$Catégorie ,grandeVilleServeur$Serveur),
        col = c("red","orange","yellow"),
        main = paste("Niveau de mise à jours des", nrow(grandeVilleServeur),
                      " serveurs des sites de ville"),
)
legend( "topleft" , c('À jour ' , "Informations masquées (vulnérabilités pas exposées)",
                      "Niveau de sécurité incertain " , "Pas à jour " ) ,
        fill = c("green","yellow","orange","red") ,cex=0.70)
```



On remarque que dans les 22 villes 11 utilisent Apache et seulement 1 seule à ses informations masquées . Les autres ne sont pas à jours .

## Version serveur

On vas maintenant observer si la version des serveurs à une influence sur la catégorie de sécurité des sites.

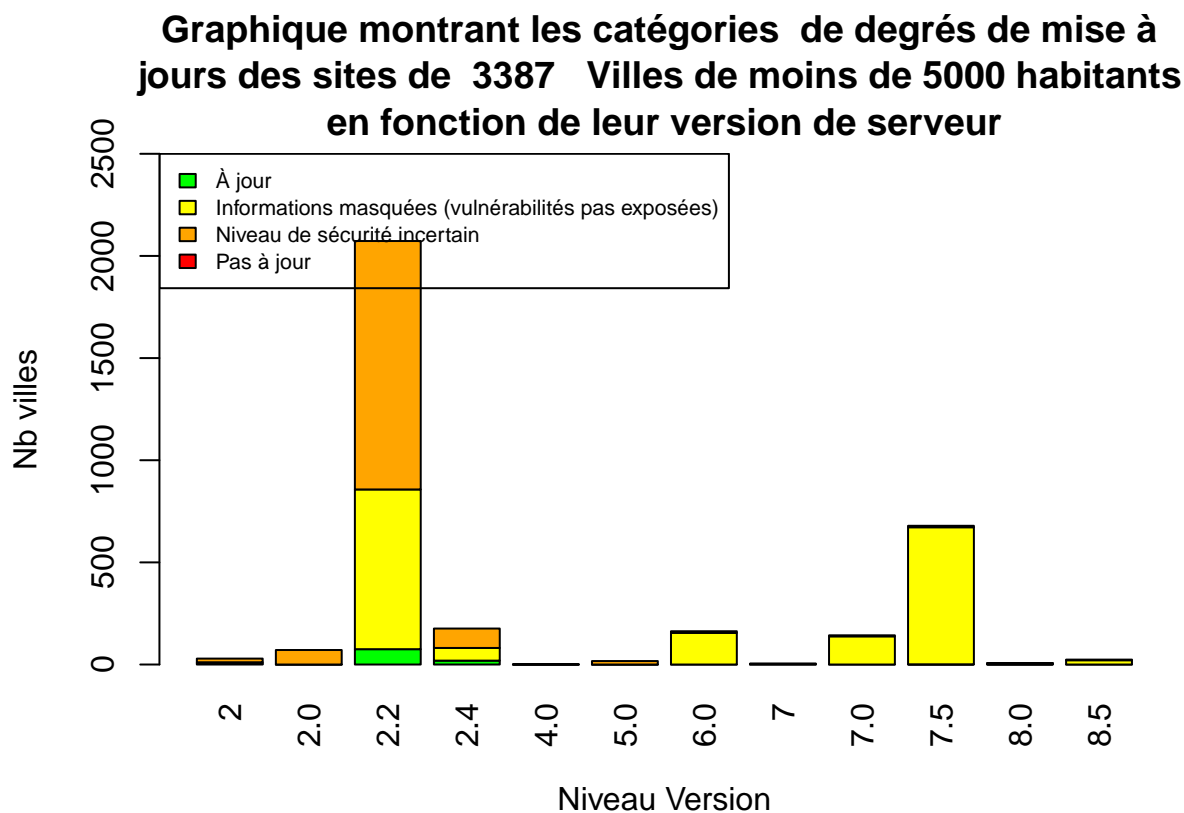
## 1. Villes de moins de 5000 habitants :

Pour les villes de moins de 5000 habitants on s'attend à ce que la versions des serveurs soient à peu près identiques à celles des grandes Villes car les sites d'état devraient certainement être hébergés sur le même type de serveurs.

```
villeVersionServeur <- filter(villeSecuInfo , 'Version du serveur' != "Inconnue" ,
Catégorie != "Inconnue")
petiteVilleVersionServeur <- filter(villeVersionServeur , Population <= 5000 , 'Version du serveur' >

barplot(table(petiteVilleVersionServeur$Catégorie,petiteVilleVersionServeur$'Version du serveur' ),
col =c("green","yellow","orange","red"),
main=paste("Graphique montrant les catégories de degrés de mise à \njours des sites de " ,
nrow(petiteVilleVersionServeur) ,
" Villes de moins de 5000 habitants \nen fonction de leur version de serveur") ,
xlab = " Niveau Version " ,
ylab="Nb villes" , las = 3 , ylim =c(0,2500))

legend( "topleft" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) ,
fill = c("green","yellow","orange","red") ,
cex=0.70)
```



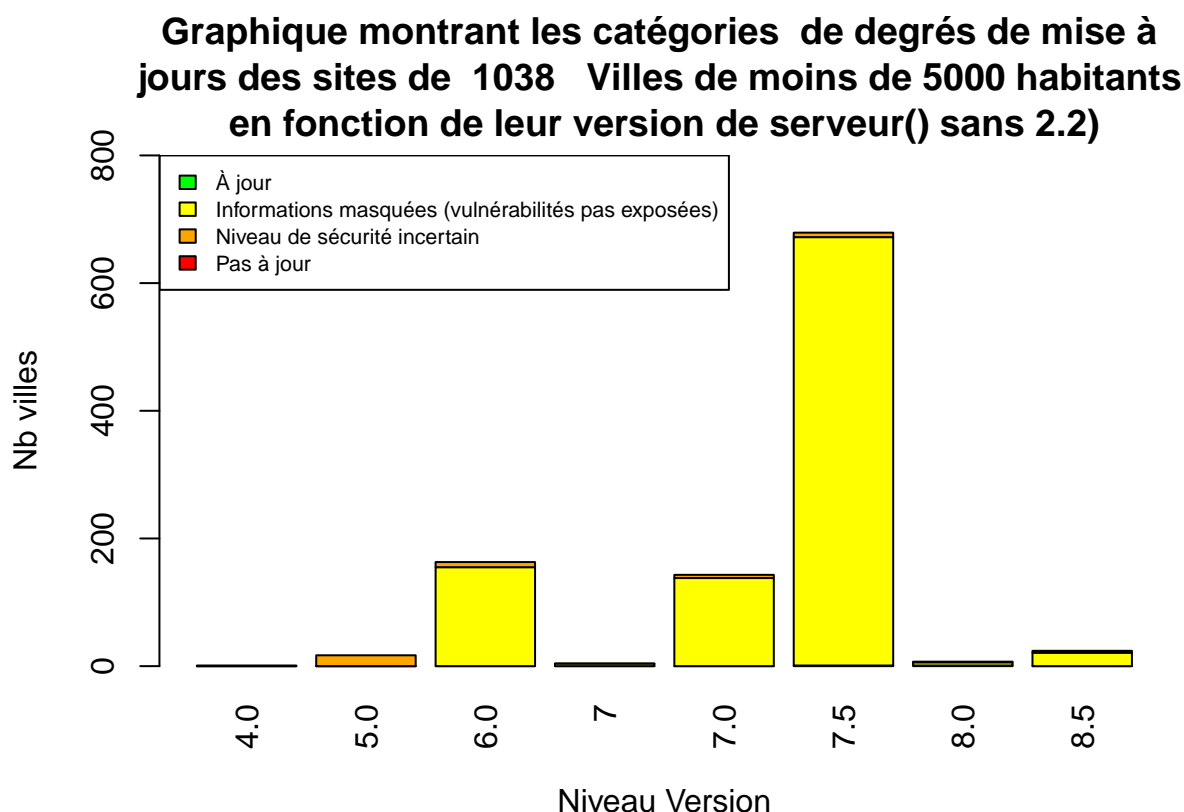
La grande majorité des serveurs sont en versions 2.2 , on observe une fois de plus un niveau de sécurité incertain pour les versions inférieures à 2.2 mais quand meme une quantité significative d'informations masquées .

On refait le diagramme en retirant la version 2.2 pour avoir une meilleur représentation du reste.

```
villeVersionServeur <- filter(villeSecuInfo , 'Version du serveur' != "Inconnue" ,
Catégorie != "Inconnue")
petiteVilleVersionServeur <- filter(villeVersionServeur , Population <= 5000 ,
'Version du serveur' > 300)

barplot(table(petiteVilleVersionServeur$Catégorie,petiteVilleVersionServeur$'Version du serveur' ),
col = c("green","yellow","orange","red") ,
main=paste("Graphique montrant les catégories de degrés de mise à \njours des sites de " ,
nrow(petiteVilleVersionServeur) ,
" Villes de moins de 5000 habitants \nen fonction de leur version de serveur() sans 2.2))" ,
xlab = " Niveau Version " ,
ylab="Nb villes" , las = 3 , ylim =c(0,800))

legend( "topleft" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) ,
fill = c("green","yellow","orange","red") ,cex=0.70)
```



Les versions 6.0 , 7.0 et 7.5 sont elles aussi remplies de sites de d'un niveau de sécurité incertains.

On essaye maintenant sans les version de serveur 2.2 et 7.5 .

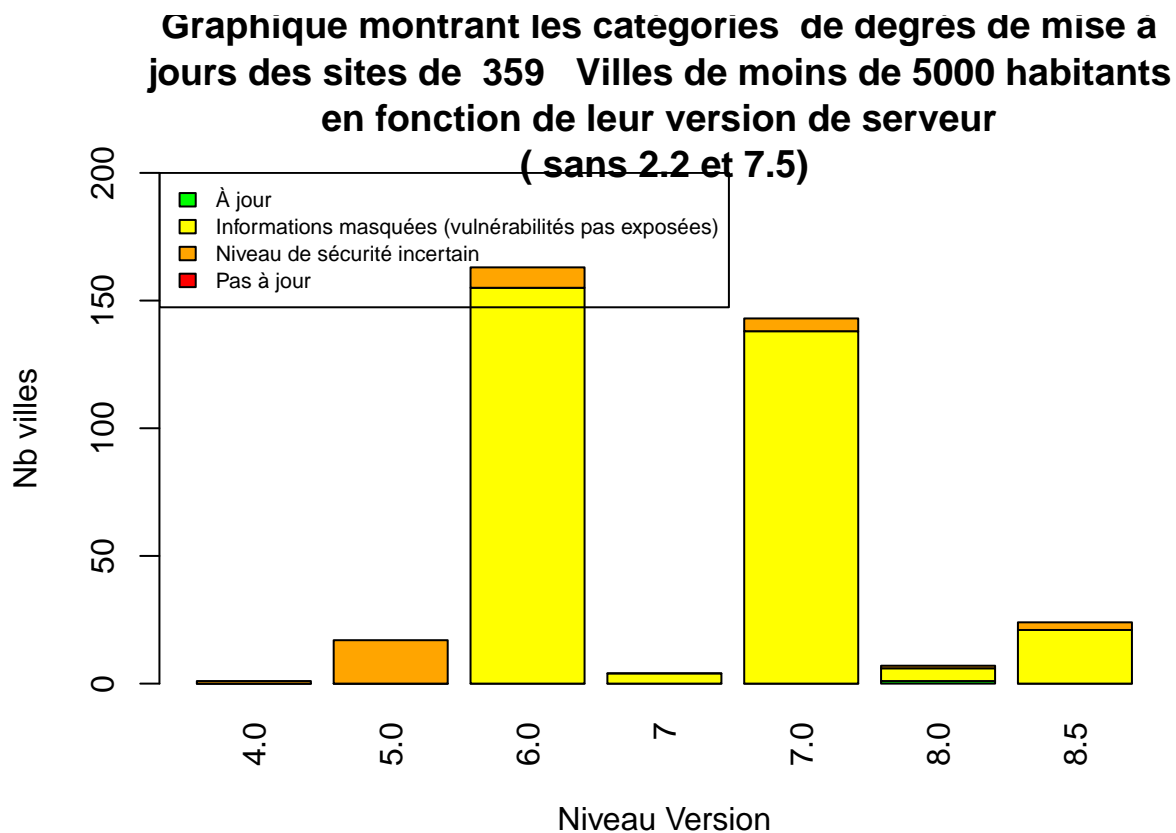
```
villeVersionServeur <- filter(villeSecuInfo , 'Version du serveur' != "Inconnue" ,
Catégorie != "Inconnue")
```

```

petiteVilleVersionServeur <- filter(villeVersionServeur , Population <= 5000 &
'Version du serveur' != "2.2" & 'Version du serveur' != "7.5" , 'Version du serveur' > 300)
barplot(table(petiteVilleVersionServeur$Catégorie,petiteVilleVersionServeur$'Version du serveur'),
col = c("green","yellow","orange","red") ,
main=paste("Graphique montrant les catégories de degrés de mise à \njours des sites de " ,
nrow(petiteVilleVersionServeur) ,
" Villes de moins de 5000 habitants \nen fonction de leur version de serveur
( sans 2.2 et 7.5)"),
xlab = " Niveau Version " ,
ylab="Nb villes" , las = 3 , ylim = c(0,200))

legend( "topleft" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) , fill = c("green","yellow","orange","red") ,
cex=0.70 )

```



On observe toujours une très faible minorité de sites aux Informations masquées .

## 2. Ville de plus de 150 000 habitants :

```

grandeVilleVersionServeur <- filter(villeVersionServeur , Population > 150000)

grandeVilleVersionServeur <- filter(villeVersionServeur , Population > 150000)

```

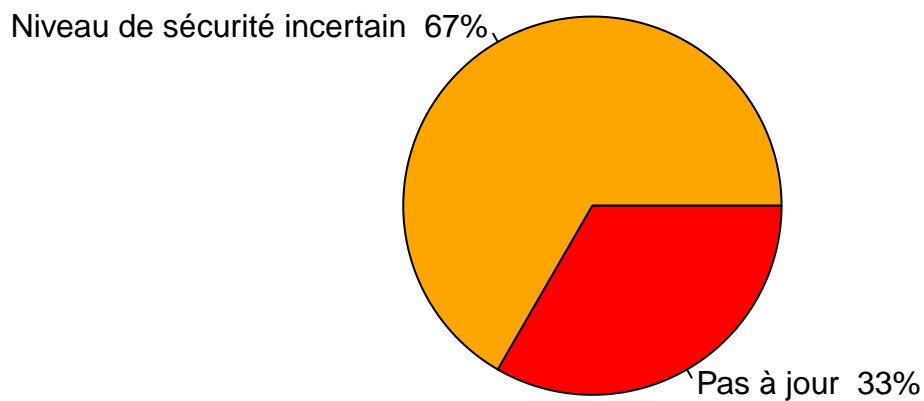
```

slices <-c(67,33)
lbls <- c("Niveau de sécurité incertain ","Pas à jour ")
pct <- pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels

pie(table(grandeVilleVersionServeur$Catégorie,grandeVilleVersionServeur$`Version du serveur`),
col =c("orange","red"),
labels = lbls ,main=paste("Graphique montrant les catégories de degrés de
mise à \njours des sites de" ,
nrow(grandeVilleVersionServeur) ,
" Villes de plus de 150 000 habitants \n en fonction deleur version de serveur"))

```

**Graphique montrant les catégories de degrés de  
mise à  
jours des sites de 3 Villes de plus de 150 000 habitants  
en fonction deleur version de serveur**



Seulement 3 grandes ville répondent aux critères, on observe que les sites sont hébergés sur une version de serveur 2.2 ; on observe aussi que 2/3 des ces villes ont un niveau de sécurité incertain , et 1/3 n'est même pas "A jour" .

On conclue donc que la version des serveurs utilisé n'a pas d'impacte sur la Catégorie des sites des petites et grandes Villes .



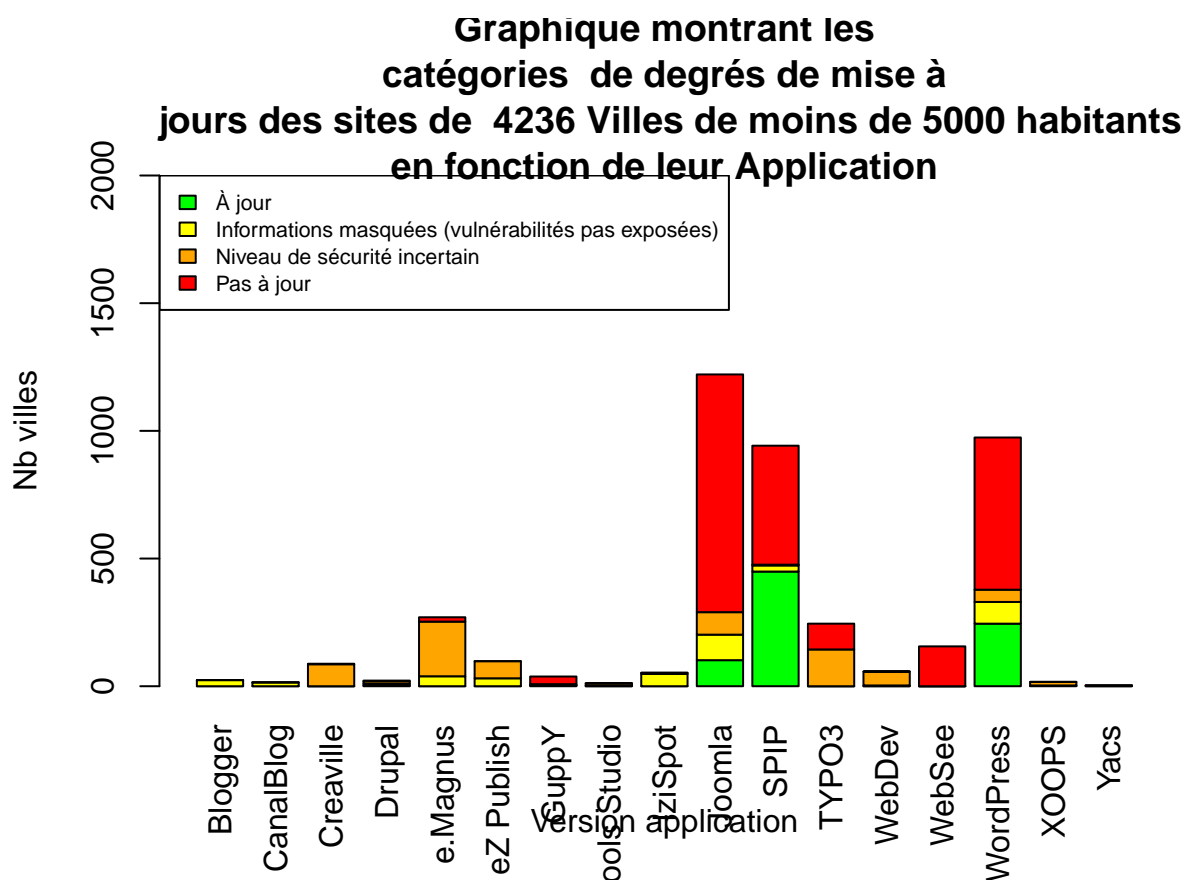
## Application

### 1. Ville de moins de 5000 habitants :

On cherche maintenant à connaître le type d'application utilisé par les sites des villes de moins de 5000 habitants .

```
par(mgp = c(3, 1, 0))
villeApp <- filter (villeSecuInfo, Catégorie != "Inconnue" , Application != "Inconnue" )
petiteVilleApp<- filter(villeApp , Population <= 5000 ,Application > 300)
barplot(table(petiteVilleApp$Catégorie,petiteVilleApp$Application ),
col = c("green","yellow","orange","red") , main=paste("Graphique montrant les
catégories de degrés de mise à\n jours des sites de "
,nrow(petiteVilleApp) ,"Villes de moins de 5000 habitants\n en fonction de leur Application ") ,
xlab = " Version application " ,
ylab="Nb villes" , las = 3 , ylim = c(0,2000) )

legend( "topleft" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) ,
fill = c("green","yellow","orange","red") ,cex=0.70)
```



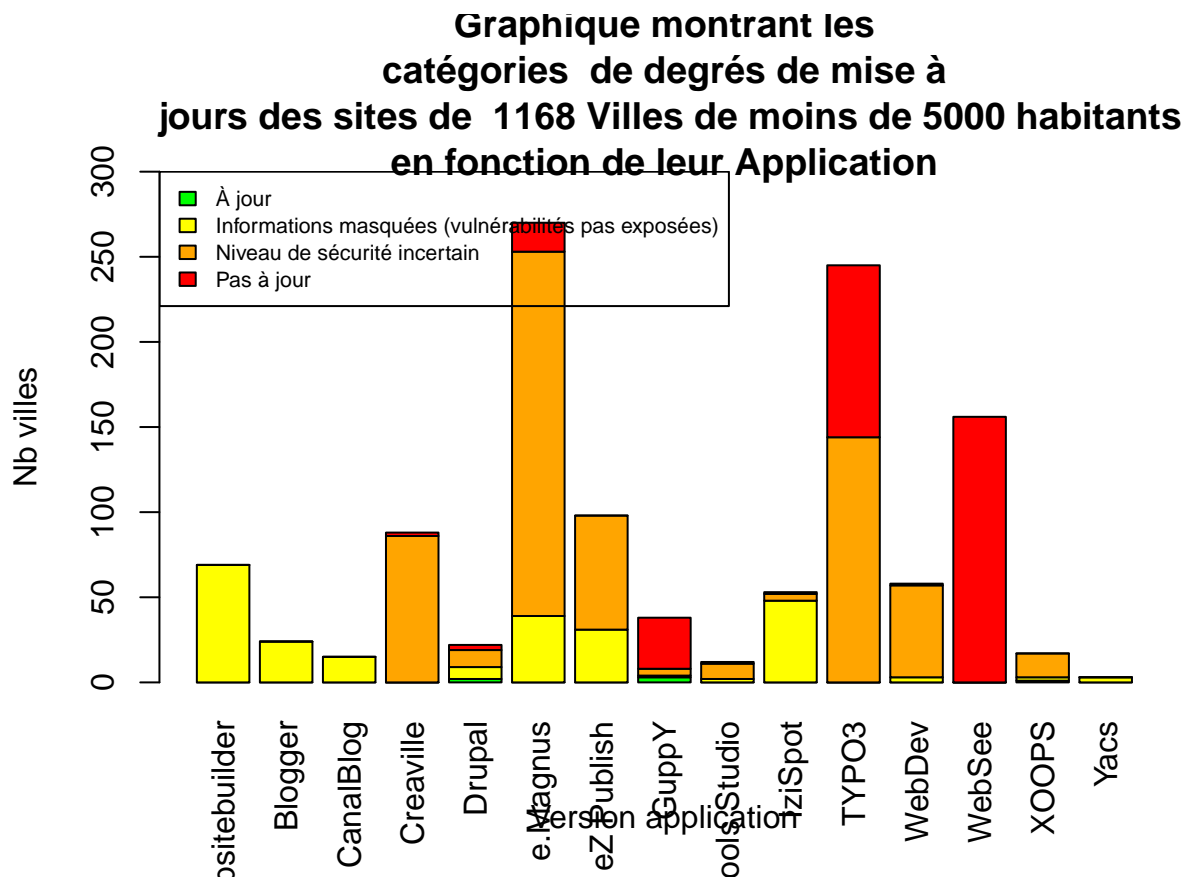
On observe que les petites villes utilisent majoritairement 3 application : - Joomla - SPIP - WordPress  
Parmis ces 3 “Applications” on remarque que plus de la moitié des sites qui les utilisent ne sont pas “A

jours”. L’“Application” contenant le plus de sites “A jour” est SPIP. Les sites fait en SPIP semblent etre majoritairement plus entenus que les autres .

On retire les “Applications” Joomla , SPIP et WordPress pour avoir un meilleur aperçus des autres “Applications” .

```
petiteVilleApp<- filter(villeApp , Population <= 5000 & Application != "Joomla" &
                        Application != "SPIP" & Application != "WordPress" )
barplot(table(petiteVilleApp$Catégorie,petiteVilleApp$Application),
col = c("green","yellow","orange","red") , main=paste("Graphique montrant les
catégories de degrés de mise à\n jours des sites de "
,nrow(petiteVilleApp) ,"Villes de moins de 5000 habitants\n en fonction de leur Application ") ,
xlab = " Version application " ,
ylab="Nb villes" , las = 3 , ylim = c(0,300) )

legend( "topleft" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) , fill = c("green","yellow","orange","red") ,
cex=0.70)
```



on observe un peu près la même chose que pour Joomla . Les sites fait avec “L’Application” WebSee ne sont pas du tout “A jour”.

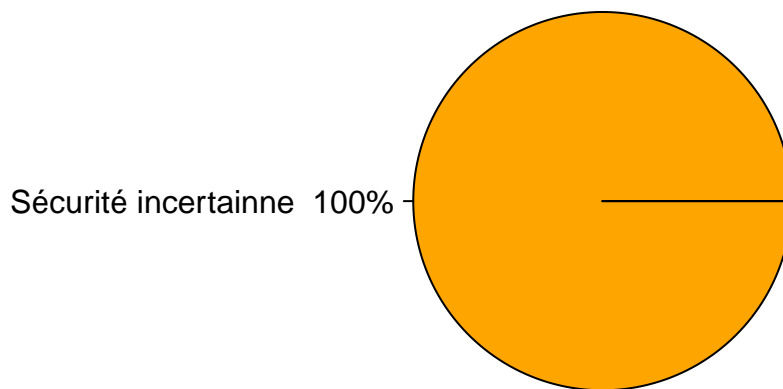
## 2. Ville de moins de 150 000 habitants :

```
villeApp <- filter (villeSecuInfo, Catégorie != "Inconnue" , Application != "Inconnue" )

grandeVilleApp <- filter(villeApp , Population > 150000)
slices <- c(100)
lbls <- c("Sécurité incertainne ")
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels

pie(table(grandeVilleApp$Catégorie,grandeVilleApp$Application) ,
col =c("orange"),
labels = lbls ,main=paste("Graphique montrant les catégories de degré
de mise à \njours des sites de ",
nrow(grandeVilleApp) ,"Villes de moins de 150000 habitants \nen fonction de leur Application"))
```

**Graphique montrant les catégories de degré  
de mise à  
jours des sites de 2 Villes de moins de 150000 habitants  
en fonction de leur Application**



Les 2 Villes correspondant aux critères utilisent TYPO3 et elles sont toutes les deux dans la catégorie “Niveau incertain”.

Le types “D’application” et la taille de la population des villes ne semblent pas avoir d’influence direct sur la catégorie de sécurité des sites webs.

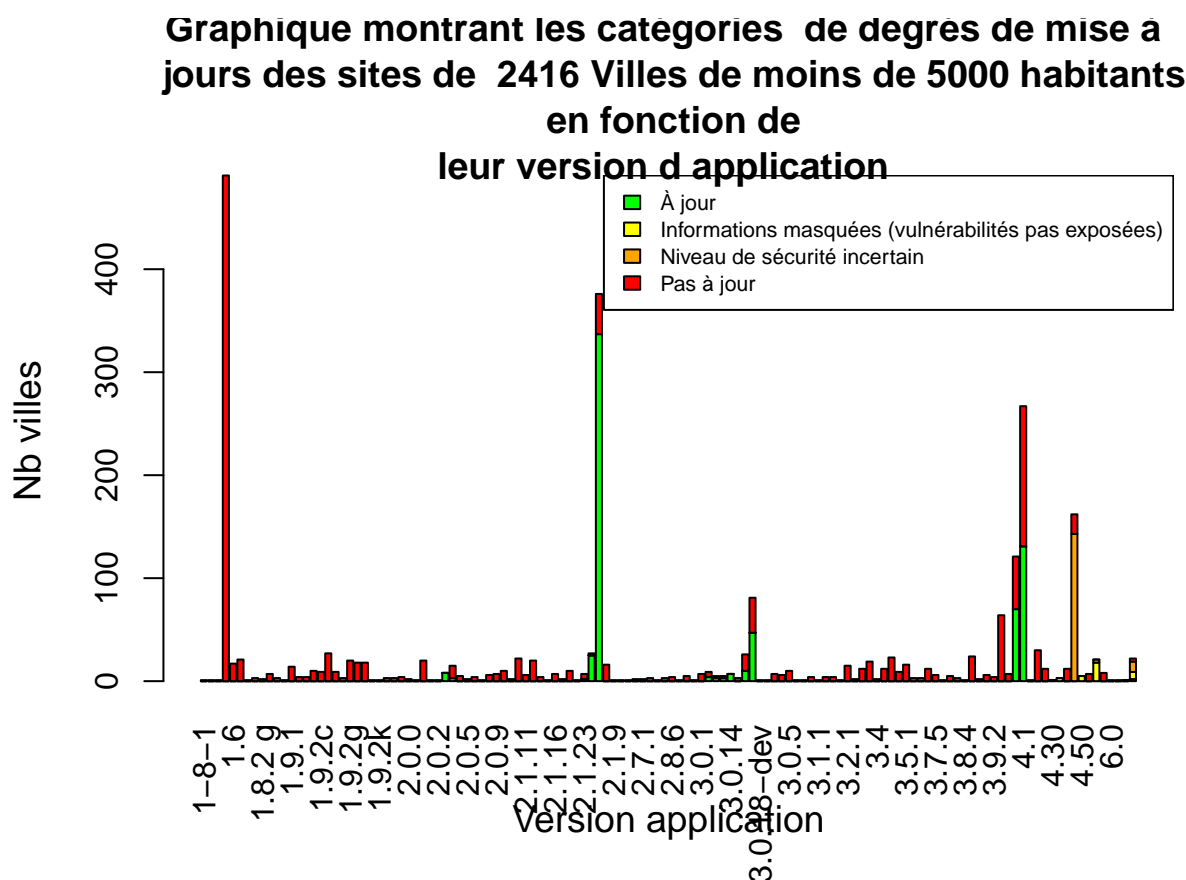
## Version Application

### 1. Ville de moins de 5000 habitants :

On cherche maintenant à savoir quelle version d'application est le plus utilisé par les villes de moins de 5000 habitants .

```
villeVersionApp <- filter (villeSecuInfo , Catégorie != "Inconnue" ,
                           'Version de l'application' != "Inconnue" )
petiteVilleVersionApp <- filter(villeVersionApp , Population <= 5000 , )
barplot(table(petiteVilleVersionApp$Catégorie,petiteVilleVersionApp$'Version de l'application'),
        col = c("green","yellow","orange","red") ,
        main=paste("Graphique montrant les catégories de degrés de mise à \n jours des sites de "
        ,nrow(petiteVilleVersionApp) ,"Villes de moins de 5000 habitants\n en fonction de
        leur version d application " ) ,
        xlab = " Version application " ,
        ylab="Nb villes" , las = 3 , cex.lab=1.2 )

legend( "topright" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
        "Niveau de sécurité incertain " , "Pas à jour " ) , fill = c("green","yellow","orange","red") ,cex=0.70)
```

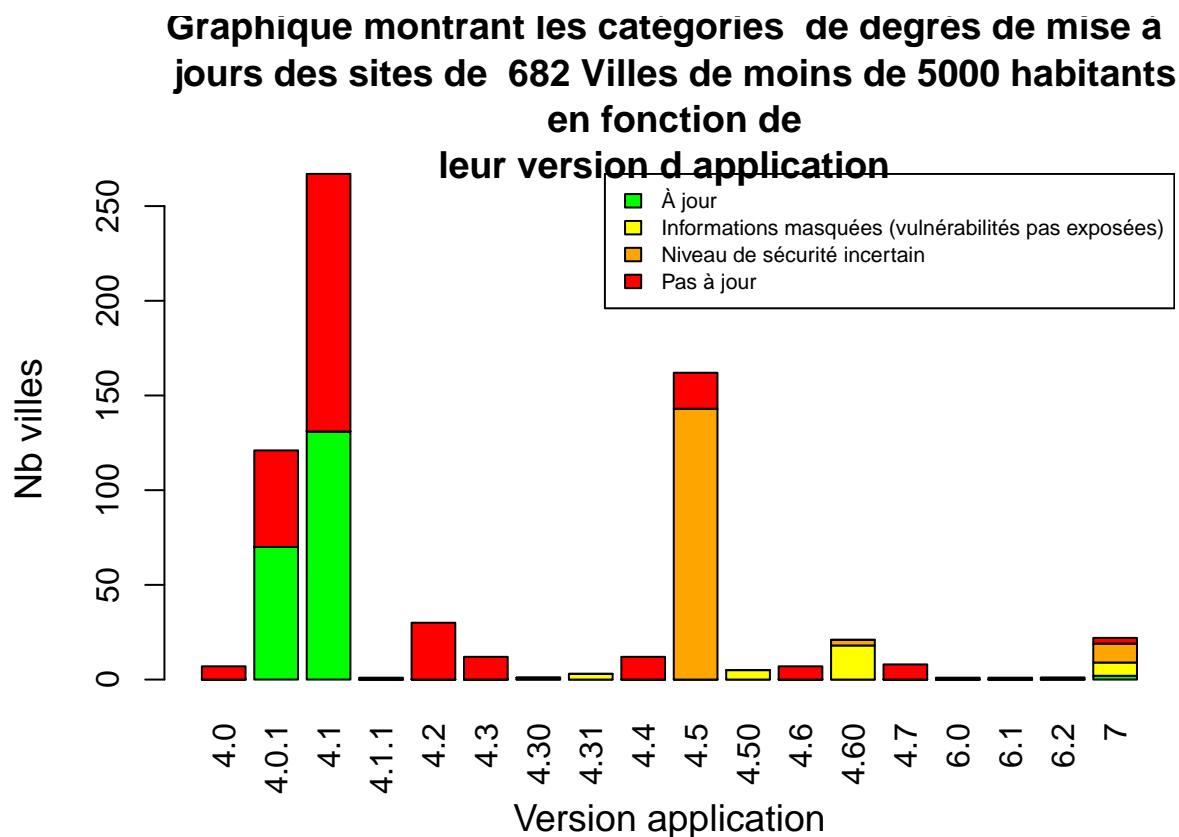


On observe que ce sont les versions 1.5 , 2.1.23 , 4.1 , 8.0.8 dev et 4.5 qui sont le plus utilisées . La version 1.8.1 est entièrement composé de site “Pas a jours” , alors que la version 2.1.23 semble etre elle la plus à jour

On cherche maintenant sans les versions 1-8-1 et 2.2.23 .

```
petiteVilleVersionApp <- filter(villeVersionApp , Population <= 5000 & 'Version de l'application' != "1-8-1" & "2.2.23")
barplot(table(petiteVilleVersionApp$Catégorie,petiteVilleVersionApp$'Version de l'application'),
col = c("green","yellow","orange","red") ,
main=paste("Graphique montrant les catégories de degrés de mise à \n jours des sites de "
,nrow(petiteVilleVersionApp) ,"Villes de moins de 5000 habitants\n en fonction de
leur version d application " ) ,
xlab = " Version application " ,
ylab="Nb villes" , las = 3 , cex.lab=1.2 )

legend( "topright" , c('À jour ' ,"Informations masquées (vulnérabilités pas exposées)",
"Niveau de sécurité incertain " , "Pas à jour " ) , fill = c("green","yellow","orange","red") ,cex=0.70)
```



La version 4.0.1 et 4.1 sont composés de près de la moitié de sites “À jour” et l’autre moitié de sites “Pas à jour” .

Les versions de l’application ne semblent pas avoir un énorme impact sur la catégorie de sécurité car la version 4.5 est composée de sites de catégorie de “sécurité incertain” et “Pas à Jour”.

## 2. Ville de plus de 150 000 habitants :

On va maintenant observer la version d’application utilisée dans les grandes villes .

```

villeVersionApp <- filter (villeSecuInfo , Catégorie != "Inconnue" ,
'Version de l'application' != "Inconnue" )
grandeVilleVersionApp <- filter(villeVersionApp , Population > 150000)

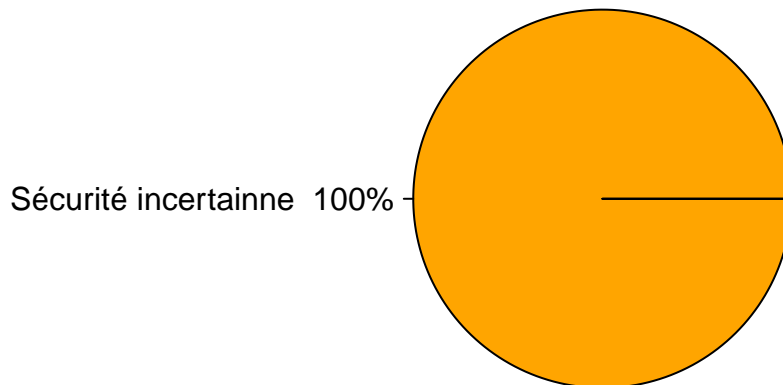
slices <-c(100)
lbls <- c("Sécurité incertaine ")
pct <- pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels

pie(table(grandeVilleVersionApp$Catégorie,grandeVilleVersionApp$'Version de l'application') ,
col =c("orange"),
labels = lbls ,main=paste("Graphique montrant les catégories de degrés de mise à\n jours des sites de
nrow(grandeVilleVersionApp) ," Villes de moins de 150 000 habitants
\n fonction de leur version d'application" ))

```

## Graphique montrant les catégories de degrés de mise à jours des sites de 1 Villes de moins de 150 000 habitants

fonction de leur version d'application



On observe que toutes utilisent la version 4.5 et qu'elles se situent dans la catégorie du niveau de sécurité incertain.

### Conclusion partie 1

La taille de la population ne semble donc pas avoir de lien direct avec la catégorie de sécurité des sites web. On pourrait penser que les sites des grandes villes (plus de 150 000 habitants) ont un meilleur service IT.

et un budget plus gors mais cela ne semble pas avoir d'influence .Il ne faut pas oublier que beaucoup de données étaient inutilisables .

## Niveau de sécurité du site en fonction de son son positionnement cartographique.

Dans ces visualisations, nous voulons savoir si le positionnement et la taille de la ville influe sur la sécurité du site. Pour cela, nous avons divisé la France en sud et nord en coupant la france sur sa ligne médiane.

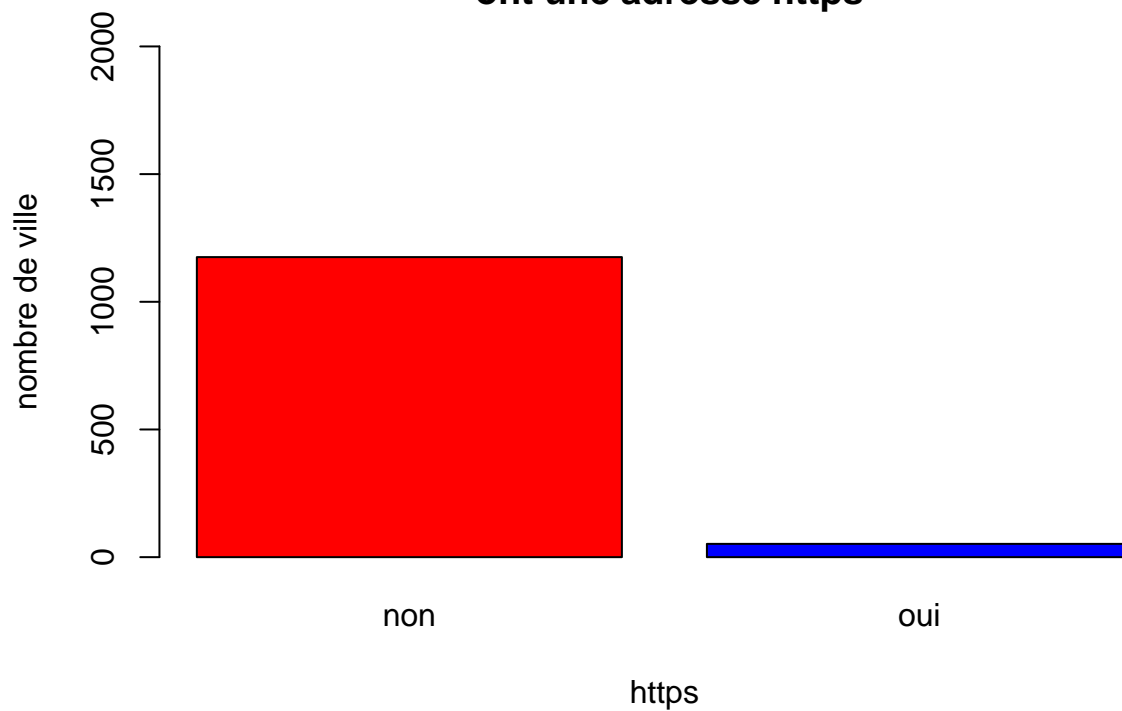
### Villes avec et sans https

#### 1. Les villes de plus de 5000 habitant qui ont une https

Dans les deux visualisations suivantes, nous voulons voir si dans les villes de plus de 5 000 habitants la différence de sécurité est élevé :

```
villeSecuInfohttpssup5000Nord <- filter(villeSecuInfo, Population > 5000
                                         & Latitude > 46.3223)
barplot(table(villeSecuInfohttpssup5000Nord$https),
        col = c("red", "blue"),
        main = paste("Les", nrow(villeSecuInfohttpssup5000Nord),
                      "villes de plus de 5000 habitants du nord qui \n ont une adresse https"),
        ylim = c(0,2000),
        xlab = "https",
        ylab = "nombre de ville"
    )
```

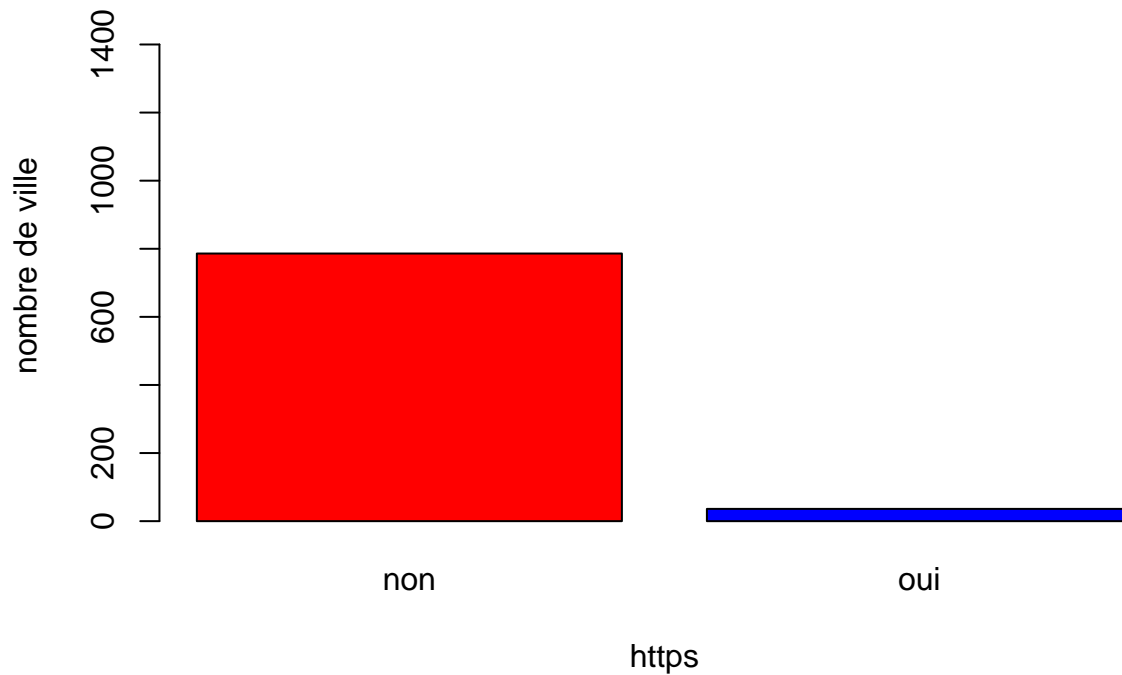
### Les 1227 villes de plus de 5000 habitants du nord qui ont une adresse https



```
villeSecuInfohttpssup5000Sud <- filter(villeSecuInfo, Population > 5000
                                         & Latitude < 46.3223)
barplot(table(villeSecuInfohttpssup5000Sud$https),
        col = c("red", "blue"),
        main = paste("Les", nrow(villeSecuInfohttpssup5000Sud),
                      "villes de plus de 5000 habitants du sud qui \nont une adresse https"),
        ylim = c(0,1500),
        xlab = "https",
        ylab = "nombre de ville")
```



## Les 822 villes de plus de 5000 habitants du sud qui ont une adresse https



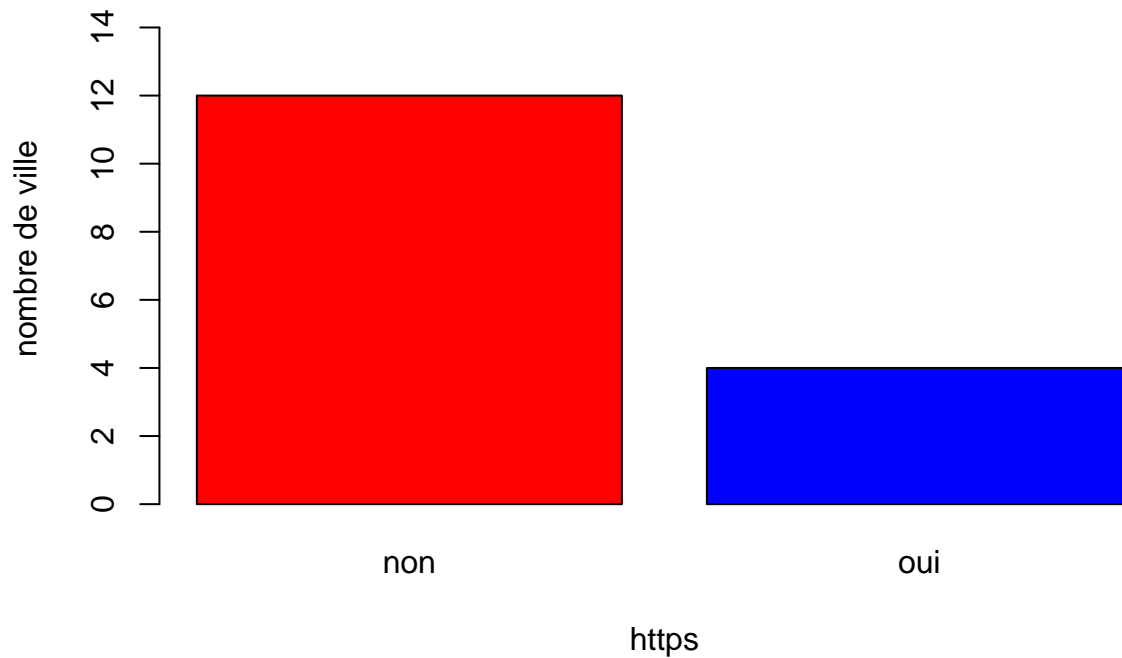
Nous Pouvons constaté que dans le nord comme dans le sud, la plupart des sites de mairie ne sont pas sécurisé, il n'y a pas de différence significative.

## 2. Les villes de plus de 150 000 habitant qui ont une https

Nous allons voir maintenant si la différence est significative pour les sites de plus de 150 000 habitans

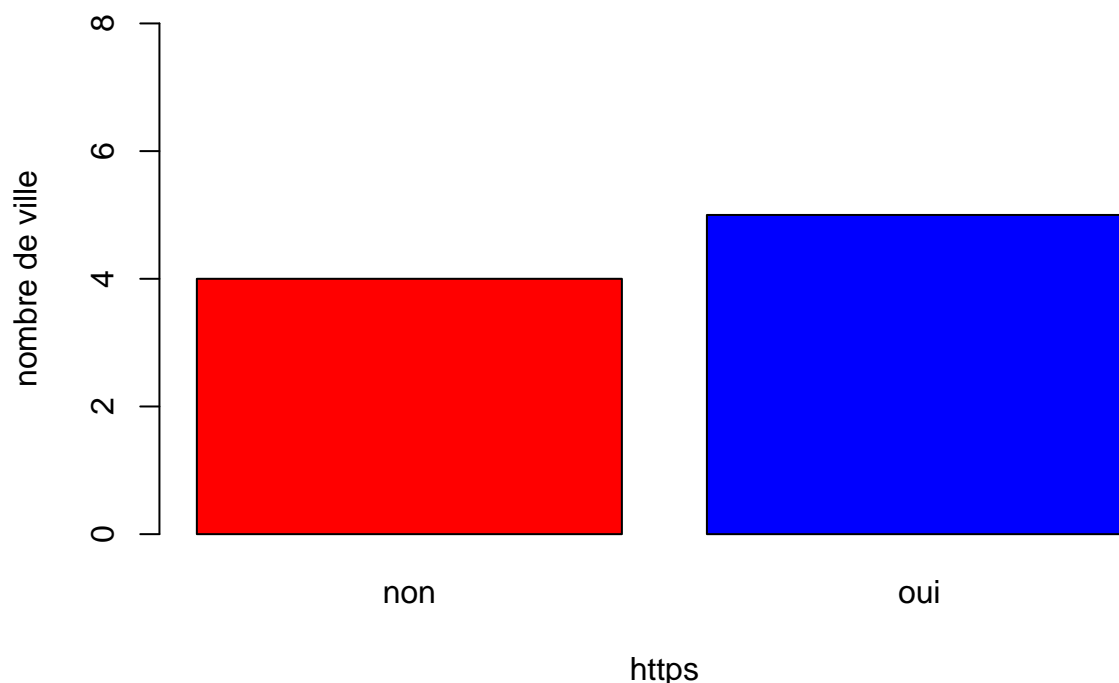
```
villeSecuInfohttpssup150000Nord <- filter(villeSecuInfo, Population > 150000
                                           & Latitude > 46.3223)
barplot(table(villeSecuInfohttpssup150000Nord$https),
        col = c("red", "blue"),
        main = paste("Les", nrow(villeSecuInfohttpssup150000Nord),
                      "villes de plus de 150 000 habitants du nord qui \n ont une adresse https"),
        ylim = c(0,15),
        xlab = "https",
        ylab = "nombre de ville"
    )
```

### Les 16 villes de plus de 150 000 habitants du nord qui ont une adresse https



```
villeSecuInfohttpssup150000Sud <- filter(villeSecuInfo, Population > 150000
                                         & Latitude < 46.3223)
barplot(table(villeSecuInfohttpssup150000Sud$https),
        col = c("red", "blue"),
        main = paste("Les", nrow(villeSecuInfohttpssup150000Sud),
                      "villes de plus de 150 000 habitants du sud qui \n ont une adresse https"),
        ylim = c(0,8),
        xlab = "https",
        ylab = "nombre de ville"
)
```

## Les 9 villes de plus de 150 000 habitants du sud qui ont une adresse https



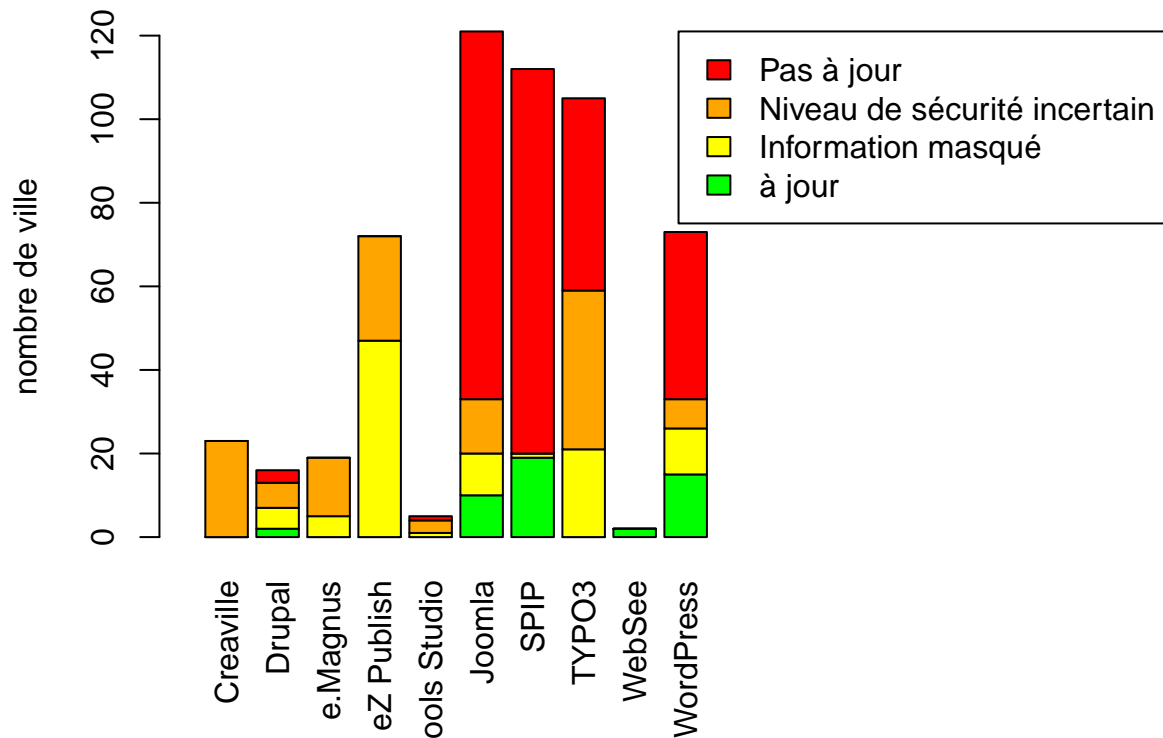
Pour les villes de plus de 150 000 habitants, nous pouvons voir que les sites des mairies du sud sont considérablement plus sécurisés que pour les villes du nord alors qu'elles sont moins nombreuses.

## Difference de sécurité en fonction de l'application utilisé

Dans ces visualisations, nous allons voir si la différence de niveau de sécurité des différentes applications entre le nord et le sud. \_\_\_\_

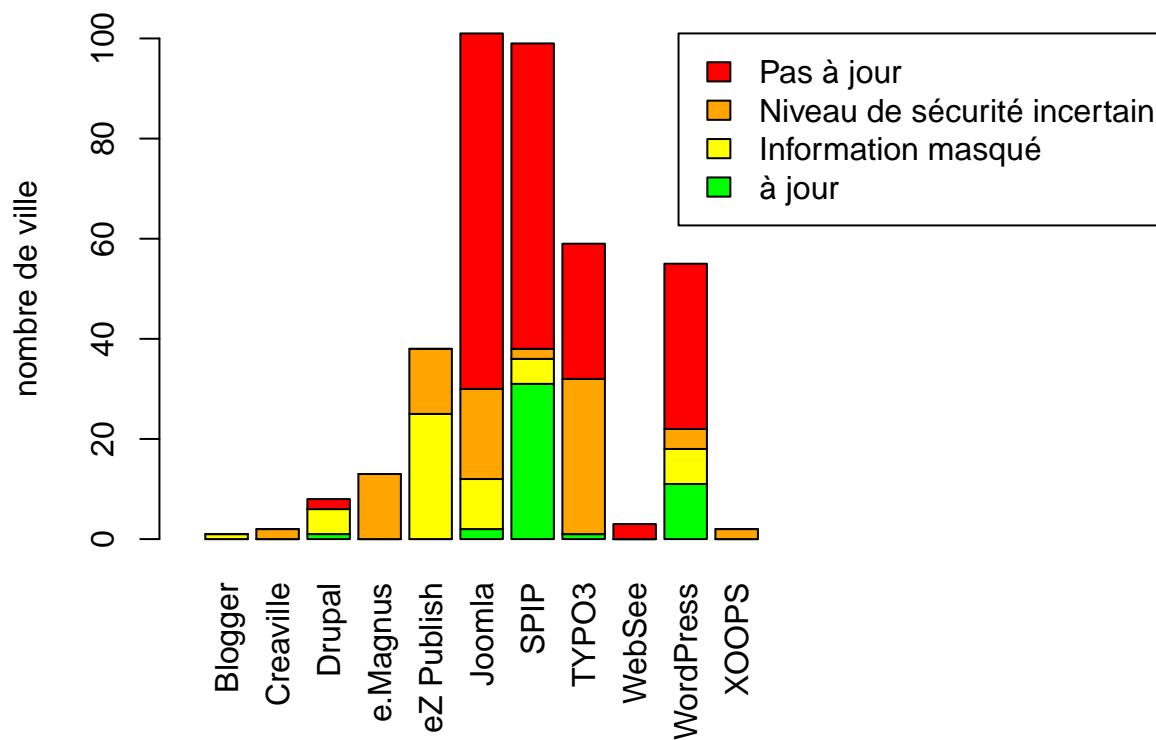
```
villeSecuInfoSup5000Nord <- filter(villeSecuInfo, Population > 5000 & Latitude > 46.3223
                                   & Application != "Inconnue" & Catégorie != "Inconnue")
barplot(table(villeSecuInfoSup5000Nord$Catégorie, villeSecuInfoSup5000Nord$Application),
col = c("green", "yellow", "orange", "red"),
main = paste("Les", nrow(villeSecuInfoSup5000Nord),
"villes de plus de 5000 habitants du nord en fonction \nde l'application qu'elles utilisent"),
ylab = "nombre de ville",
las=3,
legend.text = c("à jour", "Information masqué", "Niveau de sécurité incertain", "Pas à jour"),
args.legend = list(x = "topright", xlim = c(0, 22))
)
```

## Les 1227 villes de plus de 5000 habitants du nord en fonction de l'application qu'elles utilisent



```
villeSecuInfoSup5000Sud <- filter(villeSecuInfo, Population > 5000 & Latitude < 46.3223 &
                                Application != "Inconnue" & Catégorie != "Inconnue")
barplot(table(villeSecuInfoSup5000Sud$Catégorie, villeSecuInfoSup5000Sud$Application),
col = c("green", "yellow", "orange", "red"),
main = paste("Les", nrow(villeSecuInfoSup5000Sud),
"villes de plus de 5000 habitants du sud en fonction \nde l'application qu'elles utilisent"),
ylab = "nombre de ville",
las=3,
legend.text = c("à jour", "Information masqué", "Niveau de sécurité incertain", "Pas à jour"),
args.legend = list(x = "topright", xlim = c(0, 22))
)
```

## Les 822 villes de plus de 5000 habitants du sud en fonction de l'application qu'elles utilisent



Dans un premier temps, nous pouvons constater que dans le nord comme dans le sud, ils utilisent à peu près les mêmes applications, à l'exception de Publish et TYPO3. En suite, nous pouvons voir que globalement le niveau de mise à jour est semblable dans le nord comme dans le sud.

## Par région

Dans ce prochain graphique, nous voulions savoir quel était le département où la sécurité des sites de commune était la plus élevée. \_\_\_\_

```
parDep<-function(nDep){
  codeRegionDep <- filter(villeSecuInfo, substr('Code Insee', 0,2) == nDep)

  count <- table(unlist(codeRegionDep["https"]))
  perc <- 100*count/sum(count)
  result <- data.frame(code = sprintf('%10s', names(count)),
                      count = as.integer(count), perc = as.numeric(perc))

  #(result)

  return(result)
}
```

```

listDesDep = c('01', '02', '03', '04', '05', '06', '07', '08', '09', '10',
               '11', '12', '13', '14', '15', '16', '17', '18', '19', '20',
               '21', '22', '23', '24', '25', '26', '27', '28', '29', '30',
               '31', '32', '33', '34', '35', '36', '37', '38', '39', '40',
               '41', '42', '43', '44', '45', '46', '47', '48', '49', '50',
               '51', '52', '53', '54', '55', '56', '57', '58', '59', '60',
               '61', '62', '63', '64', '65', '66', '67', '68', '69', '70',
               '71', '72', '73', '74', '75', '76', '77', '78', '79', '80',
               '81', '82', '83', '84', '85', '86', '87', '88', '89', '90',
               '91', '92', '93', '94', '95', '2A', '2B', '971', '972', '973',
               '974', '975', '976')

prcDep<-function(listDep){
  maListPerc <- list()
  maListCode <- list()
  for ( i in 1:length(listDep)){
    maListPerc <- c( maListPerc, parDep(listDep[i])$perc[2])
    maListCode  <- c( maListCode , listDep[i])
  }

  final <- do.call(rbind, Map(data.frame, code=maListCode, perc=maListPerc))
  #
  #
  # attach(final)
  #finalOrd <- final[order("perc"),]
  #
  # detach(final)

  return(final)
}

premier15Dep <- filter(prcDep(listDesDep) , perc > 3 )

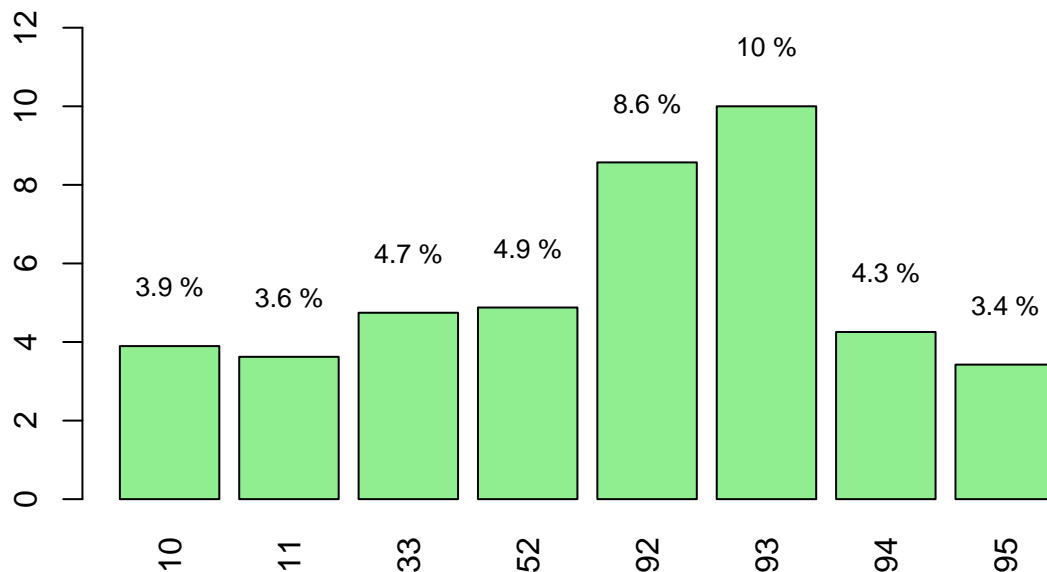
dep <- barplot(premier15Dep$perc ,
main = paste("Les ", nrow(premier15Dep),
              "départements \n avec le meilleur poucentage de sites sécurisés (HTTPS)",
col = "lightgreen",
las = 3, ylim = c(0,13),
names.arg = premier15Dep$code
)

```

```
percValeurs<-as.matrix(premier15Dep$perc)
```

```
text(dep,percValeurs+1.5,labels=as.character(paste(round(percValeurs,1),"%")) , cex = 0.8 )
```

### Les 8 départements avec le meilleur pourcentage de sites sécurisés (HTTPS)



Nous pouvons constater que c'est dans le département 93 qu'il y a le meilleur pourcentage de sites contenant un https. Mais nous pouvons encore constater que dans toute les régions la sécurité des sites de communes n'est pas assez présente quel que soit le département.

### Conclusion :

Avant l'analyse de ces données, je pensais que le nord de la France était défavorisé par rapport au sud, car les villes et les régions y sont moins riches. Cependant, l'analyse de ces données s'est montrée contradictoire. On remarque que globalement, les villes du nord ou du sud sont à peu près égales. Malgré une différence notable sur les grandes villes. En effet, les sites des grandes villes du sud se montrent plus rigoureux avec une majorité de sites https. Nous pouvons conclure après l'analyse de ces données seul pour les grandes villes, la position de la ville influe sur la sécurité du site qui lui a été attribué. \_\_\_\_

### Analyses personnelles :

#### Dahiez Ulysse

Même si nous avons globalement tous les deux travaillé sur tout le sujet, j'ai travaillé davantage sur la partie géographique, ce projet était intéressant, on peut se rendre compte du déséquilibre s'il existe entre le nord

et le sud de la France. Selon moi, la question était intéressante, car elle permet de voir qu'à peu près tous les villes et villages sont égaux quelle que soit sa position géographique. Cependant, j'aurais eu tendance à penser que les sites auraient été plus sécurisés. De plus, je pense que nous sommes dans une transition vers des sites https et que d'ici quelques années la courbe des sites ces derniers devrait considérablement monter au nord comme au sud. J'ai tout de même été surpris par la sécurité mise en place pour les grandes villes, je pense que le trafic y est élevé et qu'il serait important de les sécuriser dans un premier temps.

J'aurais aimé dans un premier temps pouvoir plus exploiter la carte, mais je ne trouvais pas de données bien explicites à montrer, c'est pourquoi nous l'avons mis en introduction. Ensuite, nous avons eu des difficultés pour afficher correctement des données en fonction du département, nous arrivions à scinder les villes et villages, mais impossible d'exploiter ces données, la typologie utilisée en cours ne fonctionnait pas (nomDeLaTable\$nomDeLaColonne). Sinon selon moi, le jeu de donnée était plutôt bon, il y avait suffisamment de données à exploiter, entre coordonnées, serveurs, niveau de sécurité, mise à jour... De plus, nous aurions pu faire les mêmes visualisations entre est et ouest, mais après quelque consultation les données n'étaient toujours assez équivalente donc pas très intéressant.

## Burdy Simon

J'ai plus travaillé sur la question de l'influence de la taille de la population sur le niveau de sécurité. J'ai fait la plupart des diagrammes pennant en compte la population . Le projet nous à permis de bien analyser et prouver nos hypothèses grâce à de multiples diagrammes. Les questions que nous sommes posées aurait peu être pu être plus précises ou plus développés , voir pertinentes.Nous aurions pu par exemple faire des hypothèses sur plusieurs groupe tels que ( très petites villes , petites villes , moyenne villes , grandes villes et très grande villes ).

L'absence de données et la multitude de données classées comme inconnues nous ont posé pas mal de problèmes car les diagrammes que nous voulions faire étaient très souvent sur une ou deux villes . Nous avons utilisé quasiment toutes les colonnes de notre jeux de données mais certaine étaient particulièrement difficile à utilisé tels que "sites" et "url" . J'ai eus de nombreuses difficulté à faire apparaitre convenablement, clairement est lisiblement les barplots . Je n'arrivais pas toujours à ne pas afficher les collones ne comportant que très peu de données .