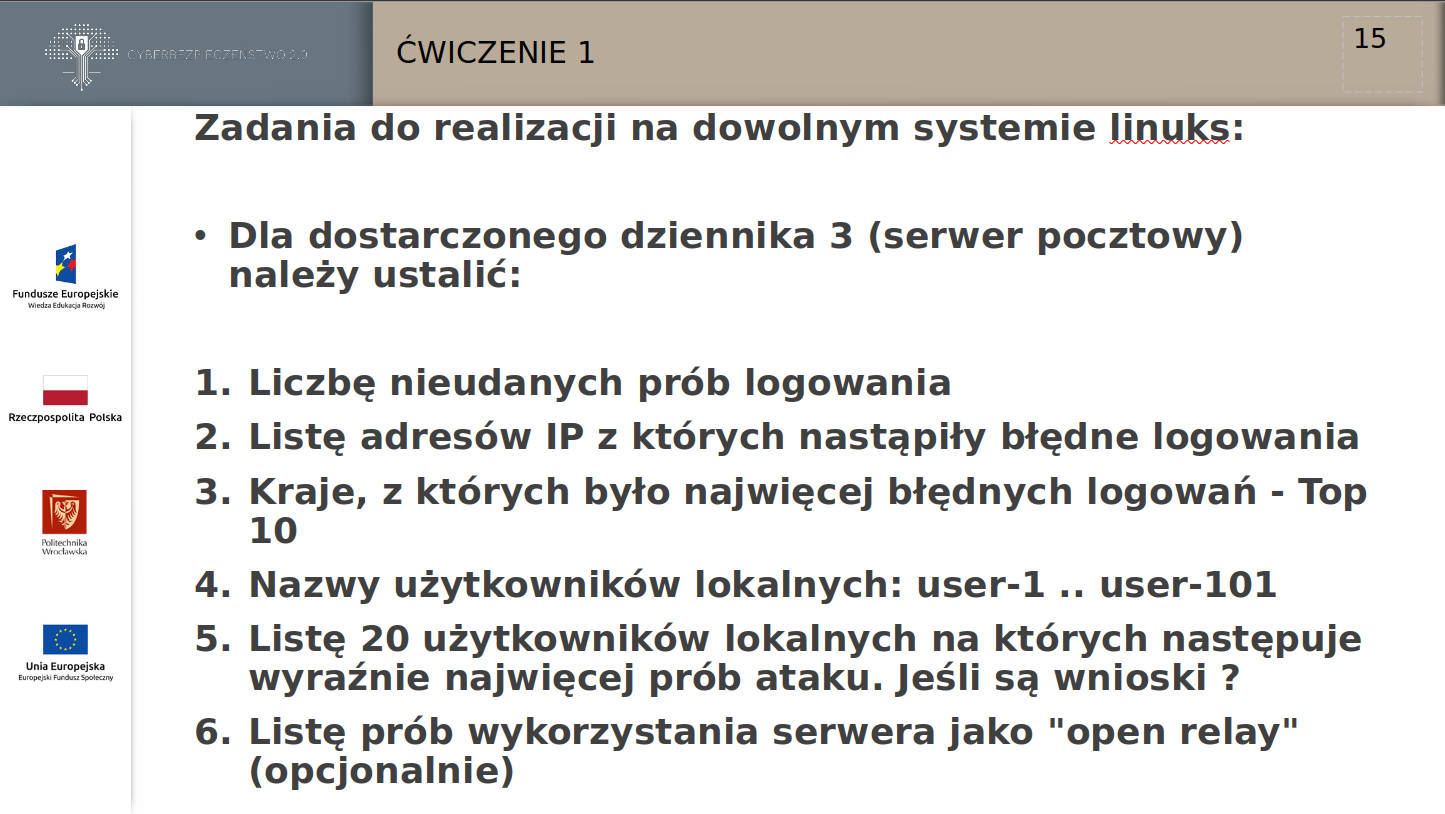
# Sprawozdanie 1 Ochrona Systemów Operacyjnych

Jonatan Kasperczak

24.03.2022

Cyberbezpieczeństwo 2022

**Dziennik 3**



Rozwiązanie wykonane w ***bash***

**Zadanie 1**

echo " Nieudane Logowania: "

grep "login authenticator failed" final.log | wc -l

Za pomocą polecenia **grep** wyszukuje frazy *“login authenticator failed”* a następnie **wc -l** liczę ile tych nieudanych logowań wystąpiło



**Zadanie 2**

echo " Lista adresów na które wykonano nieudane logowania "

grep "login authenticator failed" final.log | awk '$8 ~ /[[0-9]+.[0-9]+.[0-9]+.[0-9]+]:/ { print substr($8, 2, length($8)-3) }' | sort | uniq > most\_freq.txt

grep "login authenticator failed" final.log | awk '$9 ~ /[[0-9]+.[0-9]+.[0-9]+.[0-9]+]:/ { print substr($9, 2, length($9)-3) }' | sort |uniq >> most\_freq.txt

cat most\_freq.txt

wc -l most\_freq.txt

**awk** szuka po regex w kolumnie 8 i 9 sformatowanych odpowiednio adresów IP, poleceniem uniq usuwane są powtórzenia, i te adresy na końcu zapisywane są w pliku, plik odczytywany, i liczone linie, bo w każdej linii jest jeden adres, by podać ile było adresów IP z których były podejmowane próby logowań

**Zadanie 3**

grep "login authenticator failed" final.log | awk '$8 ~ /[[0-9]+.[0-9]+.[0-9]+.[0-9]+]:/ { print substr($8, 2, length($8)-3) }' | sort > ips.txt

grep "login authenticator failed" final.log | awk '$9 ~ /[[0-9]+.[0-9]+.[0-9]+.[0-9]+]:/ { print substr($9, 2, length($9)-3) }' | sort >> ips.txt

echo "KRAJE"

while read p; do

geoiplookup $p | awk '{$1=$2=$3=$4=""; print $0}' | sed -r 's/[ ]+/\_/g' | cut -c2- >> after\_iplookup.txt

done <ips.txt

awk '{count[$1]++} END{for (ele in count) printf "%s\t%s\n", count[ele], ele}' after\_iplookup.txt | sort -rn | sed -r 's/[\_]+/ /g'| head -10

Zapisane adresy w pliku są przez pętle poddawane **geoiplookup** który sprawdza lokalizację adresu, zapisuje do pliku, a następnie awk z pętlą **for** liczy ile jest tych samych linijek, po policzeniu sortowane są malejąco, poleceniem **sed** zamieniam znak podkreślenia spowrotem na spacje i poleceniem **head -10** wypisuje tylko 10 pierwszych linijek



**Zadanie 4**

echo "unique users"

grep "localuser" final.log | awk '{ print $5}' | cut -d '@' -f 1 | sort | uniq

Wyszukuje użytkowników lokalnych, w kolumnie 5 poleceniem **cut** dziele tekst na dwie części które oddziela znak “@” a następnie zostawiam tylko pierwszy element, sortuje i usuwa duplikaty

**Zadanie 5**

echo " Top 20 zaatakowanych użytkowników "

grep "login authenticator failed'' final.log | awk ' $13 ~ /(set\_id=)/ { print substr ($13, 9, length($13) ) }' | sort > top20user.txt

grep "login authenticator failed'' final.log | awk ' $14 ~ /(set\_id=)/ { print substr ($14, 9, length($14) ) }' | sort >> top20user.txt

uniq -c top20user.txt | sort -rn | cut -d '@' -f 1 | sed -r 's/[)+]//g' | head -20

Wyszukuje nieudane logowania, znajduje w kolumnie 13 lub 14 nazwę użytkownika na którego padła próba logowania, zapisuje do pliku. W pliku usuwa duplikaty, i od razu liczy, a następnie znów dzieli na dwie części oddzielone znakiem “@” i poleceniem **sed** usuwam ostatni znak “)” jeżeli występuje. Wypisuje 20

