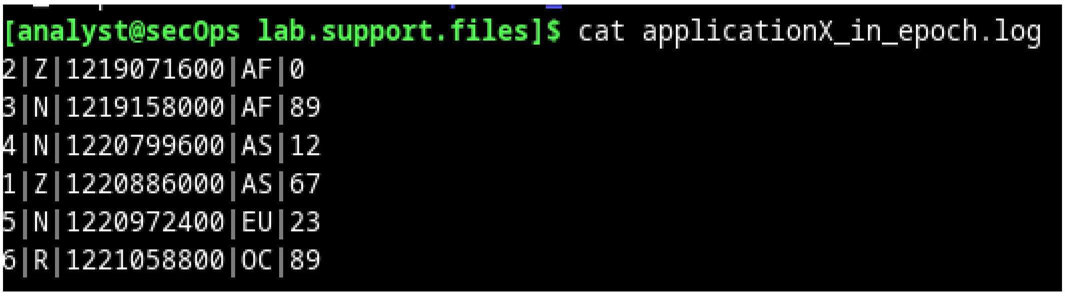
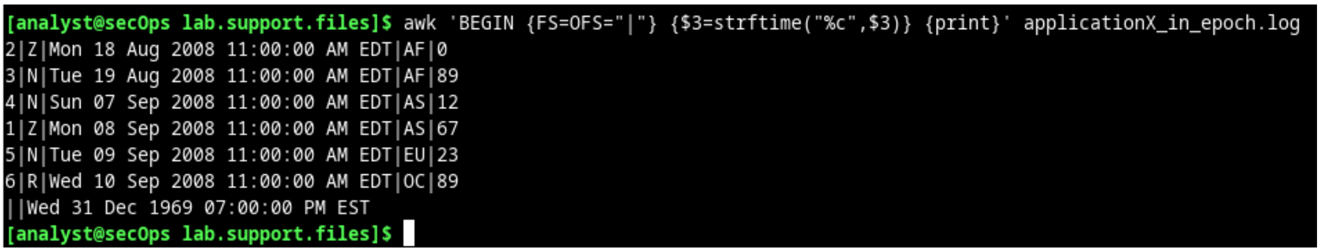
Modul_27_1_5_Convert_Data_into_a_Universal_Format

PART1

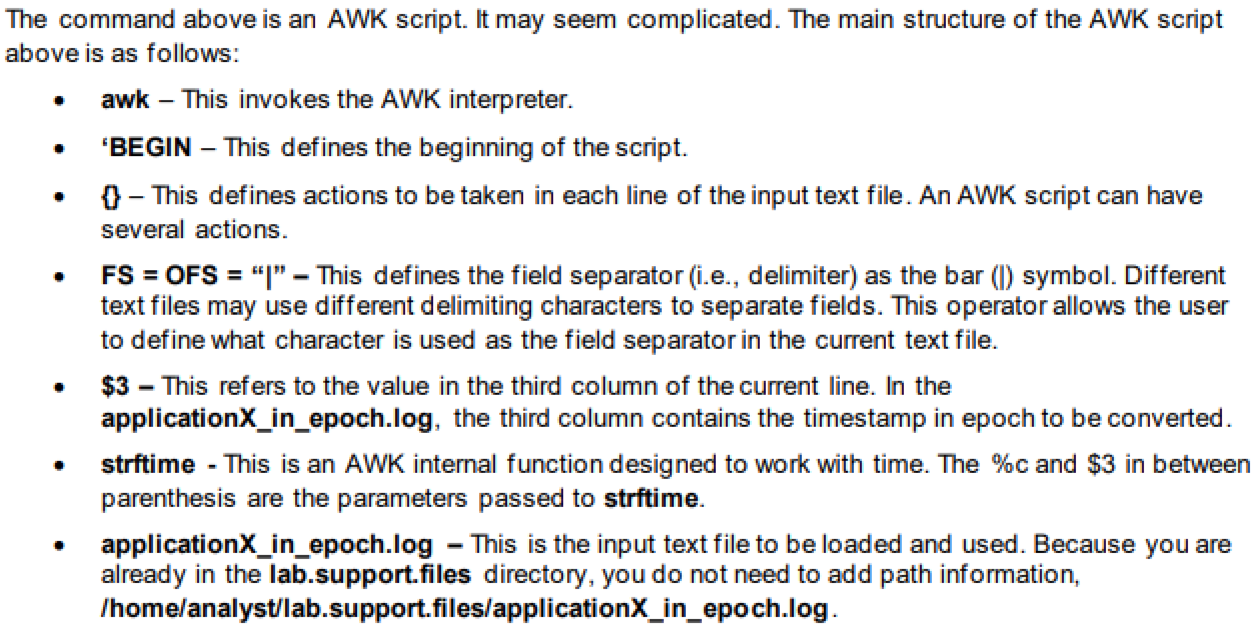
(username: analyst / password: cyberops)
c)



```
[analyst@secOps lab.support.files]$ cat applicationX_in_epoch.log
2|Z|1219071600|AF|0
3|N|1219158000|AF|89
4|N|1220799600|AS|12
1|Z|1220886000|AS|67
5|N|1220972400|EU|23
6|R|1221058800|OC|89
```



```
[analyst@secOps lab.support.files]$ awk 'BEGIN {FS=OFS="|"} {$3=strftime("%c",$3)} {print}' applicationX_in_epoch.log
2|Z|Mon 18 Aug 2008 11:00:00 AM EDT|AF|0
3|N|Tue 19 Aug 2008 11:00:00 AM EDT|AF|89
4|N|Sun 07 Sep 2008 11:00:00 AM EDT|AS|12
1|Z|Mon 08 Sep 2008 11:00:00 AM EDT|AS|67
5|N|Tue 09 Sep 2008 11:00:00 AM EDT|EU|23
6|R|Wed 10 Sep 2008 11:00:00 AM EDT|OC|89
||Wed 31 Dec 1969 07:00:00 PM EST
[analyst@secOps lab.support.files]$
```



The command above is an AWK script. It may seem complicated. The main structure of the AWK script above is as follows:

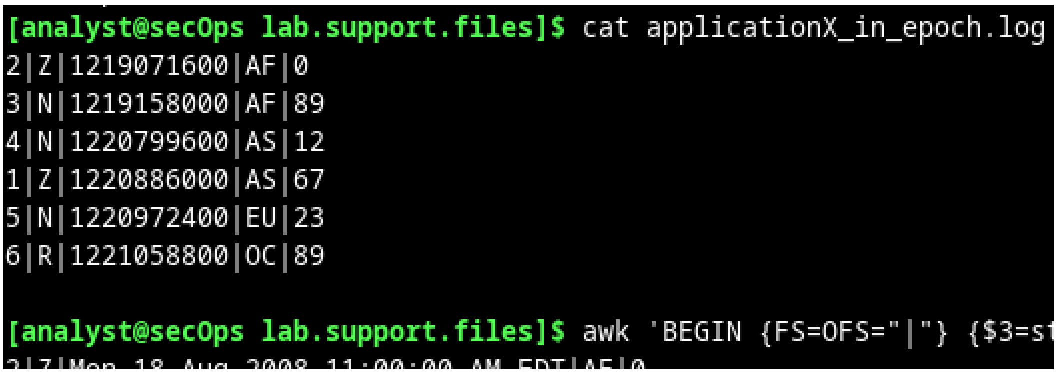
- **awk** – This invokes the AWK interpreter.
- **'BEGIN** – This defines the beginning of the script.
- **{}** – This defines actions to be taken in each line of the input text file. An AWK script can have several actions.
- **FS = OFS = "|"** – This defines the field separator (i.e., delimiter) as the bar (|) symbol. Different text files may use different delimiting characters to separate fields. This operator allows the user to define what character is used as the field separator in the current text file.
- **$3** – This refers to the value in the third column of the current line. In the **applicationX_in_epoch.log**, the third column contains the timestamp in epoch to be converted.
- **strftime** - This is an AWK internal function designed to work with time. The %c and $3 in between parenthesis are the parameters passed to **strftime**.
- **applicationX_in_epoch.log** – This is the input text file to be loaded and used. Because you are already in the **lab.support.files** directory, you do not need to add path information, **/home/analyst/lab.support.files/applicationX_in_epoch.log**.

Were the Unix Epoch timestamps converted to Human Readable format? Were the other fields modified? Explain.

- Ano, byly. Jediné co se změnilo byl 3 sloupec, který byl konvertován do normální podoby

Compare the contents of the file and the printed output. Why is there the line, ||Wed 31 Dec 1969 07:00:00 PM EST?

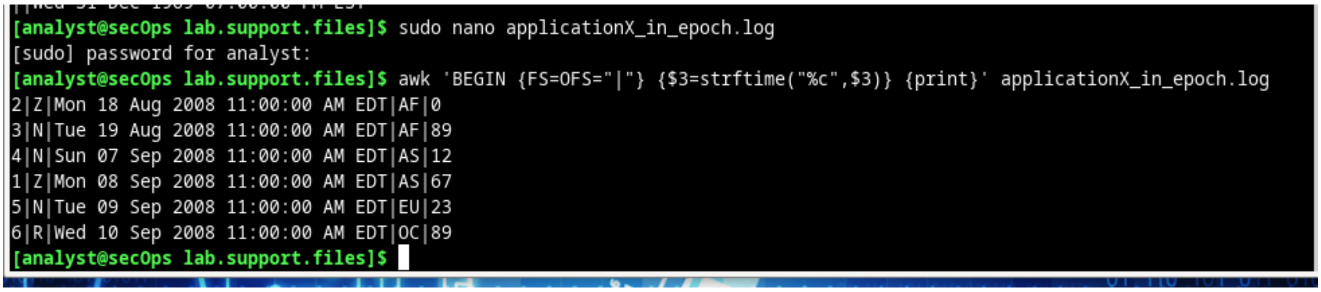
- pravděpodobně, protože je uložený new-line v originálním soubor, který se bral defaultně jako začátek?



```
[analyst@secOps lab.support.files]$ cat applicationX_in_epoch.log
2|Z|1219071600|AF|0
3|N|1219158000|AF|89
4|N|1220799600|AS|12
1|Z|1220886000|AS|67
5|N|1220972400|EU|23
6|R|1221058800|OC|89

[analyst@secOps lab.support.files]$ awk 'BEGIN {FS=OFS="|"} {$3=st
2|Z|Mon 18 Aug 2008 11:00:00 AM EDT|AF|0
```

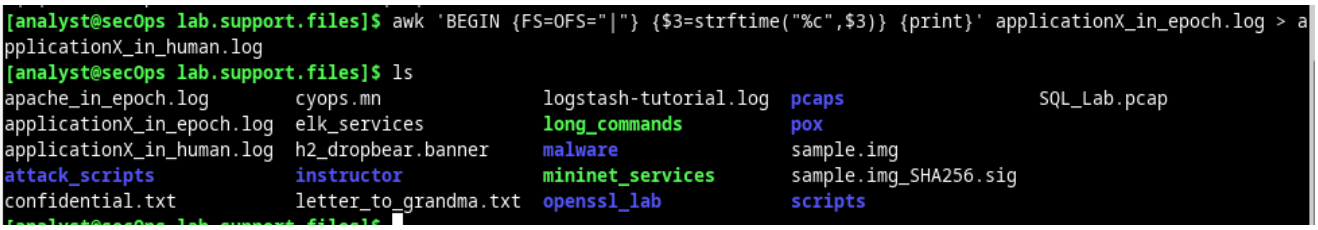
d)



```
||Wed 31 Dec 1969 07:00:00 PM EST
[analyst@secOps lab.support.files]$ sudo nano applicationX_in_epoch.log
[sudo] password for analyst:
[analyst@secOps lab.support.files]$ awk 'BEGIN {FS=OFS="|"} {$3=strftime("%c",$3)} {print}' applicationX_in_epoch.log
2|Z|Mon 18 Aug 2008 11:00:00 AM EDT|AF|0
3|N|Tue 19 Aug 2008 11:00:00 AM EDT|AF|89
4|N|Sun 07 Sep 2008 11:00:00 AM EDT|AS|12
1|Z|Mon 08 Sep 2008 11:00:00 AM EDT|AS|67
5|N|Tue 09 Sep 2008 11:00:00 AM EDT|EU|23
6|R|Wed 10 Sep 2008 11:00:00 AM EDT|OC|89
[analyst@secOps lab.support.files]$ █
```

Smazali jsme new-line, takže output už je správný

e)



```
[analyst@secOps lab.support.files]$ awk 'BEGIN {FS=OFS="|"} {$3=strftime("%c",$3)} {print}' applicationX_in_epoch.log > a
pplicationX_in_human.log
[analyst@secOps lab.support.files]$ ls
apache_in_epoch.log      cyops.mn             logstash-tutorial.log  pcaps                  SQL_Lab.pcap
applicationX_in_epoch.log  elk_services       long_commands          pox
applicationX_in_human.log  h2_dropbear.banner  malware               sample.img
attack_scripts           instructor           mininet_services       sample.img_SHA256.sig
confidential.txt         letter_to_grandma.txt  openssl_lab          scripts
[analyst@secOps lab.support.files]$ █
```

What was printed by the command above? Is this expected?

- ano, jelikož jsme output commandu vložili do souboru applicationX_in_human.log

Part 2

```
[analyst@secOps lab.support.files]$ cat apache_in_epoch.log
198.51.100.213 - - [1219071600] "GET
/twiki/bin/edit/Main/Double_bounce_sender?topicparent=Main.ConfigurationVariables
HTTP/1.1" 401 12846
198.51.100.213 - - [1219158000] "GET
/twiki/bin/rdiff/TWiki/NewUserTemplate?rev1=1.3&rev2=1.2 HTTP/1.1" 200 4523
198.51.100.213 - - [1220799600] "GET /mailman/listinfo/hsdivision HTTP/1.1" 200 6291
198.51.100.213 - - [1220886000] "GET /twiki/bin/view/TWiki/WikiSyntax HTTP/1.1" 200
7352
198.51.100.213 - - [1220972400] "GET /twiki/bin/view/Main/DCCAndPostFix HTTP/1.1" 200
5253
198.51.100.213 - - [1221058800] "GET
/twiki/bin/oops/TWiki/AppendixFileSystem?template=oopsmore&m1=1.12&m2=1.12 HTTP/1.1"
200 11382
```

The Apache Log file above contains six entries which record events related to the Apache web server. Each entry has seven fields. The fields are delimited by a space:

- The first column contains the IPv4 address, **198.51.100.213**, of the web client placing the request.
- The second and third columns are not used and a "-" character is used to represent no value.
- The fourth column contains the timestamp in Unix Epoch time, for example **[1219071600]**.
- The fifth column contains text with details about the event, including URLs and web request parameters. All six entries are HTTP GET messages. Because these messages include spaces, the entire field is enclosed with quotes.
- The sixth column contains the HTTP status code, for example **401**.
- The seventh column contains the size of the response to the client (in bytes), for example **12846**.

a)

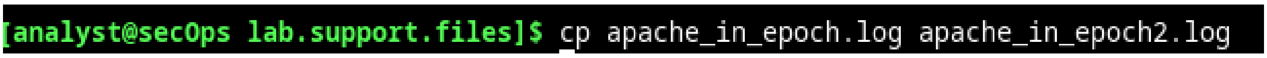
In the context of timestamp conversion, what character would work as a good delimiter character for the Apache log file above?

- mezera

  How many columns does the Apache log file above contain?
- 7

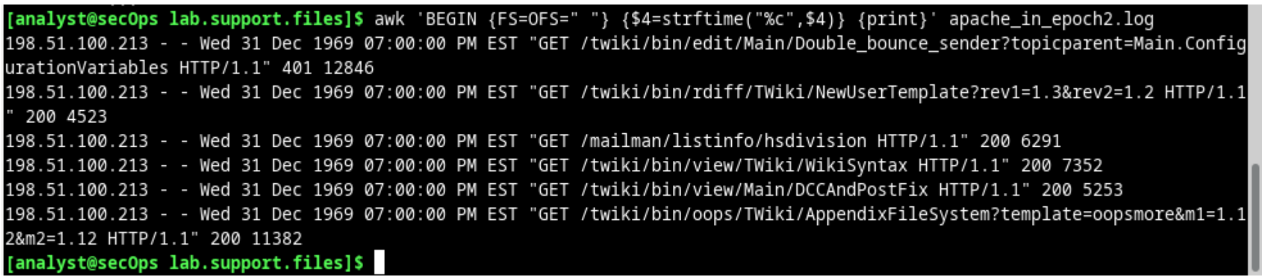
  In the Apache log file above, what column contains the Unix Epoch Timestamp?
- 4

  b)



```
[analyst@secOps lab.support.files]$ cp apache_in_epoch.log apache_in_epoch2.log
```

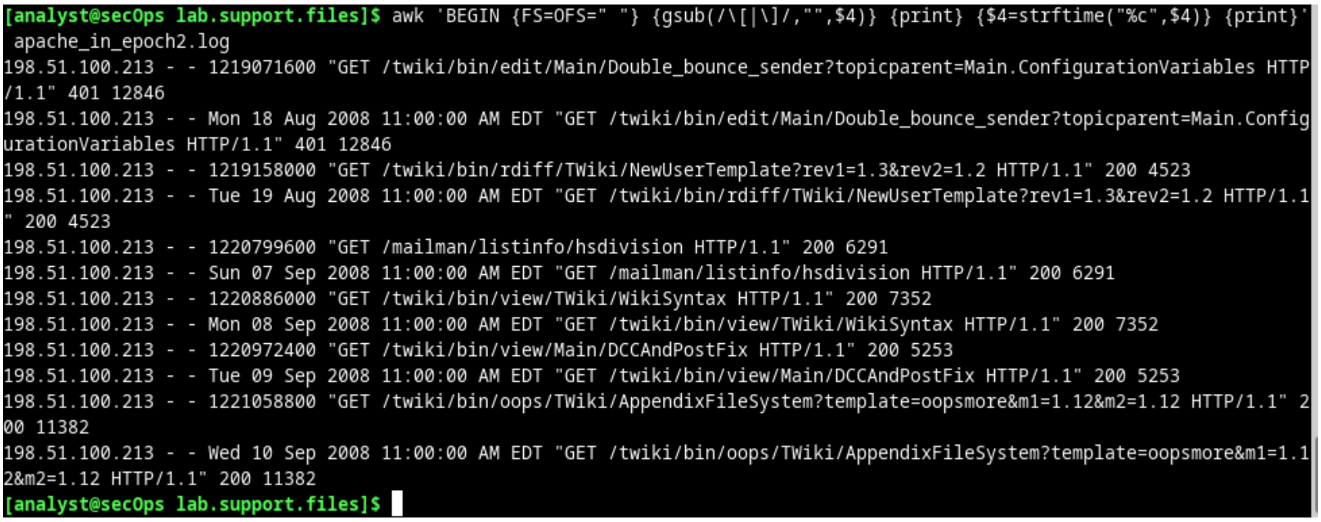
c + d)



```
[analyst@secOps lab.support.files]$ awk 'BEGIN {FS=OFS=" "} {$4=strftime("%c",$4)} {print}' apache_in_epoch2.log
198.51.100.213 - - Wed 31 Dec 1969 07:00:00 PM EST "GET /twiki/bin/edit/Main/Double_bounce_sender?topicparent=Main.Config
urationVariables HTTP/1.1" 401 12846
198.51.100.213 - - Wed 31 Dec 1969 07:00:00 PM EST "GET /twiki/bin/rdiff/TWiki/NewUserTemplate?rev1=1.3&rev2=1.2 HTTP/1.1
" 200 4523
198.51.100.213 - - Wed 31 Dec 1969 07:00:00 PM EST "GET /mailman/listinfo/hsdivision HTTP/1.1" 200 6291
198.51.100.213 - - Wed 31 Dec 1969 07:00:00 PM EST "GET /twiki/bin/view/TWiki/WikiSyntax HTTP/1.1" 200 7352
198.51.100.213 - - Wed 31 Dec 1969 07:00:00 PM EST "GET /twiki/bin/view/Main/DCCAndPostFix HTTP/1.1" 200 5253
198.51.100.213 - - Wed 31 Dec 1969 07:00:00 PM EST "GET /twiki/bin/oops/TWiki/AppendixFileSystem?template=oopsmore&m1=1.1
2&m2=1.12 HTTP/1.1" 200 11382
[analyst@secOps lab.support.files]$
```

- output se změnil v timestampu jak bylo určeno, ale jejich obsah je chybný
- ne, pravděpodobně kvůli závorkám [], protože je asi potřeba čistý text

e)



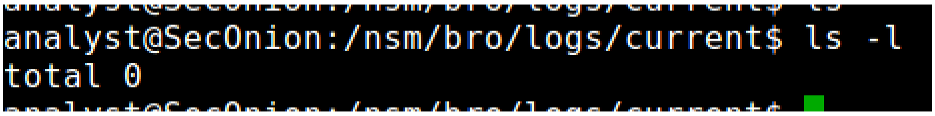
```
[analyst@secOps lab.support.files]$ awk 'BEGIN {FS=OFS=" "} {gsub(/\[|\]/,"",$4)} {print} {$4=strftime("%c",$4)} {print}'
 apache_in_epoch2.log
198.51.100.213 - - 1219071600 "GET /twiki/bin/edit/Main/Double_bounce_sender?topicparent=Main.ConfigurationVariables HTTP
/1.1" 401 12846
198.51.100.213 - - Mon 18 Aug 2008 11:00:00 AM EDT "GET /twiki/bin/edit/Main/Double_bounce_sender?topicparent=Main.Config
urationVariables HTTP/1.1" 401 12846
198.51.100.213 - - 1219158000 "GET /twiki/bin/rdiff/TWiki/NewUserTemplate?rev1=1.3&rev2=1.2 HTTP/1.1" 200 4523
198.51.100.213 - - Tue 19 Aug 2008 11:00:00 AM EDT "GET /twiki/bin/rdiff/TWiki/NewUserTemplate?rev1=1.3&rev2=1.2 HTTP/1.1
" 200 4523
198.51.100.213 - - 1220799600 "GET /mailman/listinfo/hsdivision HTTP/1.1" 200 6291
198.51.100.213 - - Sun 07 Sep 2008 11:00:00 AM EDT "GET /mailman/listinfo/hsdivision HTTP/1.1" 200 6291
198.51.100.213 - - 1220886000 "GET /twiki/bin/view/TWiki/WikiSyntax HTTP/1.1" 200 7352
198.51.100.213 - - Mon 08 Sep 2008 11:00:00 AM EDT "GET /twiki/bin/view/TWiki/WikiSyntax HTTP/1.1" 200 7352
198.51.100.213 - - 1220972400 "GET /twiki/bin/view/Main/DCCAndPostFix HTTP/1.1" 200 5253
198.51.100.213 - - Tue 09 Sep 2008 11:00:00 AM EDT "GET /twiki/bin/view/Main/DCCAndPostFix HTTP/1.1" 200 5253
198.51.100.213 - - 1221058800 "GET /twiki/bin/oops/TWiki/AppendixFileSystem?template=oopsmore&m1=1.12&m2=1.12 HTTP/1.1" 2
00 11382
198.51.100.213 - - Wed 10 Sep 2008 11:00:00 AM EDT "GET /twiki/bin/oops/TWiki/AppendixFileSystem?template=oopsmore&m1=1.1
2&m2=1.12 HTTP/1.1" 200 11382
[analyst@secOps lab.support.files]$
```

f)

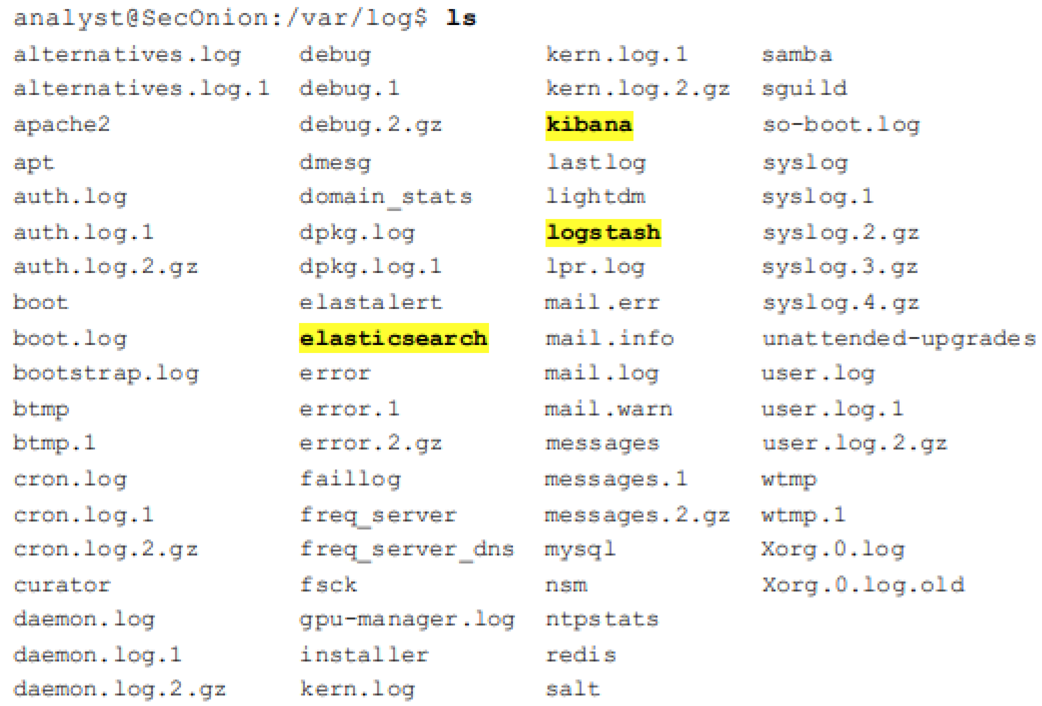
Was the script able to properly convert the timestamps this time? Describe the output

- Ano, teď máme dva printy první nekonvertlý s odstraněnými závorky a druhý už konvertlý

Part3



```
analyst@SecOnion:/nsm/bro/logs/current$ ls
analyst@SecOnion:/nsm/bro/logs/current$ ls -l
total 0
analyst@SecOnion:/nsm/bro/logs/current$
```



```
analyst@SecOnion:/var/log$ ls
alternatives.log    debug              kern.log.1      samba
alternatives.log.1  debug.1            kern.log.2.gz   sguild
apache2             debug.2.gz         kibana          so-boot.log
apt                 dmesg              lastlog         syslog
auth.log            domain_stats       lightdm         syslog.1
auth.log.1          dpkg.log           logstash        syslog.2.gz
auth.log.2.gz       dpkg.log.1         lpr.log         syslog.3.gz
boot                elastalert         mail.err        syslog.4.gz
boot.log            elasticsearch      mail.info       unattended-upgrades
bootstrap.log       error              mail.log        user.log
btmp                error.1            mail.warn       user.log.1
btmp.1              error.2.gz         messages        user.log.2.gz
cron.log            faillog            messages.1      wtmp
cron.log.1          freq_server        messages.2.gz   wtmp.1
cron.log.2.gz       freq_server_dns    mysql           Xorg.0.log
curator             fsck               nsm             Xorg.0.log.old
daemon.log          gpu-manager.log    ntpstats
daemon.log.1        installer          redis
daemon.log.2.gz     kern.log           salt
```

For each one of the tools listed above, describe the function, importance, and placement in the security analyst workflow.

Elasticsearch

- Function: "search and analytics engine" (built on Apache Lucene). ukládá a indexuje logy z různých zdrojů

- Importance: Efektivně ukládá a vyhledává velké objemy logů. Podporuje analýzu bezpečnostních dat v reálném čase. Umožňuje bezpečnostním týmům odhalovat anomálie a korelace mezi událostmi.
- Placement: úložistě a vyhledávání (logy z firewallů, IDS/IPS systémů... jsou indexovány v Elasticsearch -> umožňuje rychlejší vyhledávání)

Logstash

- Function: nástroj pro zpracování dat, který shromažďuje, upravuje a přeposílá logy z různých zdrojů
- Importance: Čistí a strukturuje surová bezpečnostní data pro lepší analýzu. Obohacuje logy o další informace (např. reputace IP adres, geolokace).
- Placement: zpracování a sběr dat (předzpracování logů (před elsticsearch))

Kibana

- Function: nástroj pro vizualizaci a tvorbu dashboardů -> Poskytuje vizuální analýzu v reálném čase, přehledné dashboardy a možnosti alertingu.
- Importance: Pomáhá bezpečnostním analytikům vizualizovat hrozby. Umožňuje efektivní reakci na incidenty díky identifikaci vzorců útoků. Podporuje proaktivní vyhledávání hrozeb pomocí dotazů a automatických upozornění.
- Placement: Vizualizace a analýza (plus monitoring)