

**FACULDADE DE TECNOLOGIA SENAC DE GOIÁS**  
**CURSO GESTÃO DA TECNOLOGIA DA INFORMAÇÃO**  
**LABORATÓRIO DE REDES DE COMPUTADORES**

**JOÃO NETTO NATAL PINHEIRO**  
**JOÃO PAULO NASCIMENTO OLIVEIRA**  
**PAULO ROBERTO VIEIRA**  
**RUBEM DE OLIVEIRA VIEIRA**

**Tratamento de arquivo de log HTTP**

Goiânia  
2018

**JOÃO NETTO NATAL PINHEIRO**  
**JOÃO PAULO NASCIMENTO OLIVEIRA**  
**PAULO ROBERTO VIEIRA**  
**RUBEM DE OLIVEIRA VIEIRA**

**Tratamento de arquivo de log HTTP**

Trabalho apresentado como requisito parcial de nota da disciplina Laboratório de Redes de Computadores do curso Gestão da Tecnologia da Informação da Faculdade de Tecnologia SENAC.

Orientação: Fernando Tsukahara

## **1- O que são arquivos de log?**

Arquivos log são arquivos criados e mantidos por softwares que descrevem o seu funcionamento ou uma parte dele. Uma vez criado, esse arquivo passa a ser alimentado com dados para, caso houver necessidade, ser analisado posteriormente. Um arquivo log pode ser gerado e mantido a partir de vários eventos, tais como: alterações em arquivos, acesso a recursos e, no caso do foco deste trabalho, conexões com um servidor web.

Em um arquivo log genérico, pode-se encontrar informações do momento do evento, quem realizou, o que foi feito, entre outras. Tais informações são bastantes úteis para desenvolvedores de softwares e gestores de TI em suas ações e decisões.

Arquivos log devem ser fáceis de se interpretar para uma pessoa com conhecimento técnico. Muitas vezes, os softwares permitem a personalizações da geração desses arquivos, podendo o administrador aumentar ou diminuir o nível de detalhamento.

Como os arquivos log são gerados automaticamente, seja por uma interação ou por um erro do sistema, eles podem crescer tanto em tamanho que prejudicam o funcionamento do sistema. Um dos papéis do administrador é analisar a criação desses arquivos e se certificar de que não atrapalhará o sistema em vez de ajudar. Os logs são em formato de texto e podem possuir, tranquilamente, mais de 500Mb, então analisar, compactar, copiar ou, até mesmo apagar esses arquivos deve ser comum.

A análise dos logs é algo extremamente útil e pode ser considerado uma importante ferramenta na tomada das decisões de um gestor ou desenvolvedor. Para tal, esses arquivos, na maioria das vezes, são bem protegidos principalmente contra a escrita, evitando que sejam alterados indevidamente.

## **2- O formato de log do HTTP**

Vários aplicativos e serviços geram arquivos de log mas trataremos, exclusivamente, do arquivo log gerado por um servidor HTTP Apache. O arquivo em questão é acrescido a toda nova conexão com o servidor, seja um envio de formulário ou o carregamento de uma imagem. Um servidor HTTP também gera log de erros, porém vamos analisar somente o log de acessos.

### **2.1- Formatos padronizados de log**

O administrador do sistema tem total autonomia para alterar o formato padrão do arquivo log, como já foi mencionado, porém o Apache possui uma configuração padrão permanece ativa caso não haja alteração.

O servidor apache2 instalado e configurado no CentOS7, por padrão, apresenta os campos:

```
%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-agent}i\"
```

Onde:

%h – Representa o endereço IP do cliente que fez a requisição ao servidor. Dependendo da configuração do servidor, esse campo pode ser substituído pelo nome do hostname porém não é uma opção recomendada visto que diminui a performance.

%l – Indica a identidade RFC 1413 do cliente determinado pelo identd na máquina do cliente. Quando está representado pelo hífen, quer dizer que essa informação não está disponível. Esta informação não é tão confiável e quase nunca é usada, exceto em casos especiais.

%u – Mostra o usuário que requisitou a página http. Caso a página não seja protegida, essa informação não será útil e será representada por um hífen.

%t – Representa a data referente ao término do processamento da requisição. O formato padrão é [dia/mês/ano:hora:minuto:segundo zona] onde o dia é representado por dois dígitos; o mês por três letras; o ano por quatro dígitos; a hora, minuto e segundo por dois dígitos e a zona por um símbolo de + ou – mais quatro dígitos.

\"%r\" - Esse campo representa o que foi requisitado do servidor http. Dessa linha, retira-se muitas informações importantes como o tipo de requisição, o que foi requisitado e o tipo de protocolo usado.

%>s – Esse campo refere-se ao código de status retornado pelo servidor. Tal código é bastante útil porque mostra a resposta da requisição feita pelo cliente.

%b – Esse campo representa o tamanho do objeto retornado para o cliente, não incluindo os cabeçalhos. Se nenhum objeto for retornado, essa informação será exibida como “-”.

\"%{Referer}i\" - Esse campo mostra o cabeçalho da requisição HTTP.

\"%{User-agent}i\" - Esse campo representa o tipo de cliente, podendo ser um Browser comum ou outro software (gerenciado e-mail, por exemplo).

## **2.2 – Quais dados podem ser extraídos do arquivo de log HTTP?**

Além das informações explícitas no próprio arquivo log, várias outras informações podem ser extraídas com o tratamento dessas informações. Os arquivos de log possuem dados, estes combinados de forma gerencial, podem ser transformados em informação e posteriormente conhecimento, baseando-se nestes, uma empresa pode decidir quais estratégias utilizar na sua tomada de decisão, ou seja, gestão baseada em conhecimento. Esse tipo de decisão vem crescendo de forma significativa no mundo empresarial. Por exemplo, fazendo uso do IP, pode-se determinar, mesmo que sem muita precisão, qual o país de origem de determinado IP. Com isso, decisões gerenciais podem ser tomadas como por exemplo, implementar um novo idioma no website.

As informações de data e hora também são muito importantes pois podem determinar uma faixa de tempo em que o servidor é mais requisitado, alocando melhor os recursos computacionais. A análise do browser também permite, ao gestor, identificar quais navegadores mais acessam o servidor e melhorar a experiência para tais usuários.

Até mesmo os códigos de respostas podem ser úteis para saber como o servidor se comporta em determinada hora com determinados acessos.

## **3 – Arquivos de log escolhidos para o teste**

Os arquivos usados nos testes foram disponibilizados pelo professor Fernando Pirkel Tsukahara. Existe três arquivos com tamanho distintos, o programa a ser entregue no projeto integrador, fará a leitura, extração dos dados e análise gerenciais dos mesmos, segue os links para tais arquivos:

<https://portalpbh.pbh.gov.br/logs/access.log>

<http://www.devqa.robotec.co.il/apache/logs/access.log>

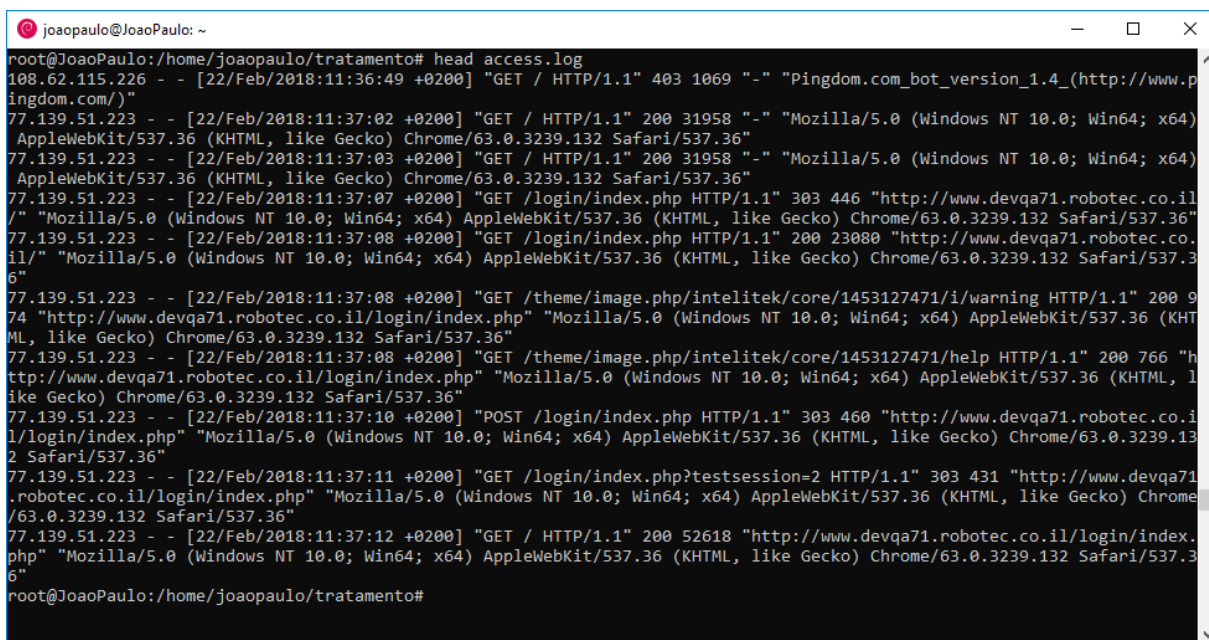
<http://www.almhuetten-raith.at/apache-log/access.log>

## 4 – Tratamento dos dados

No tratamento dos dados, os arquivos log foram separados em arquivos que contém o IP, a data e o browser separadamente. Para efetuar essa operação, usa-se o comando “awk” com alguns parâmetros. Finalmente, para juntar os três arquivos em um só e configurar um separador que o software Java reconheça, utiliza-se o comando “paste”.

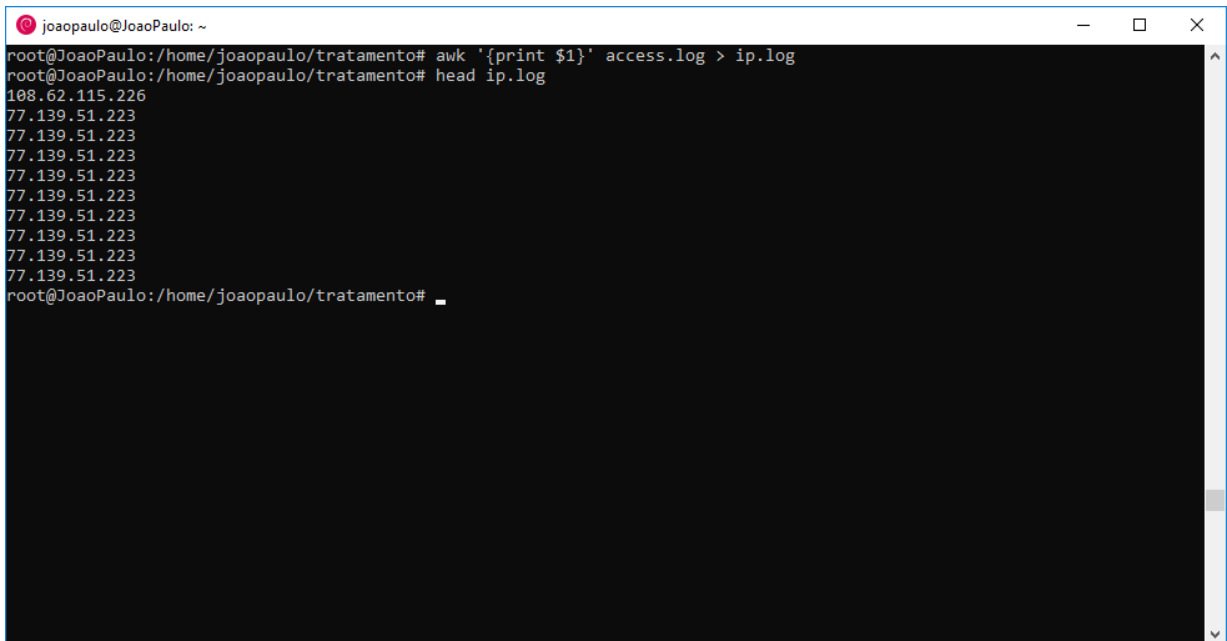
Para essa descrição, foi utilizado o arquivo log do segundo link disponibilizado, cujo nome será “access.log”. Vale lembrar que o procedimento pode ser feito para quaisquer arquivos log.

1º - Para confirmar o padrão do arquivo, usa-se o comando “head access.log”.



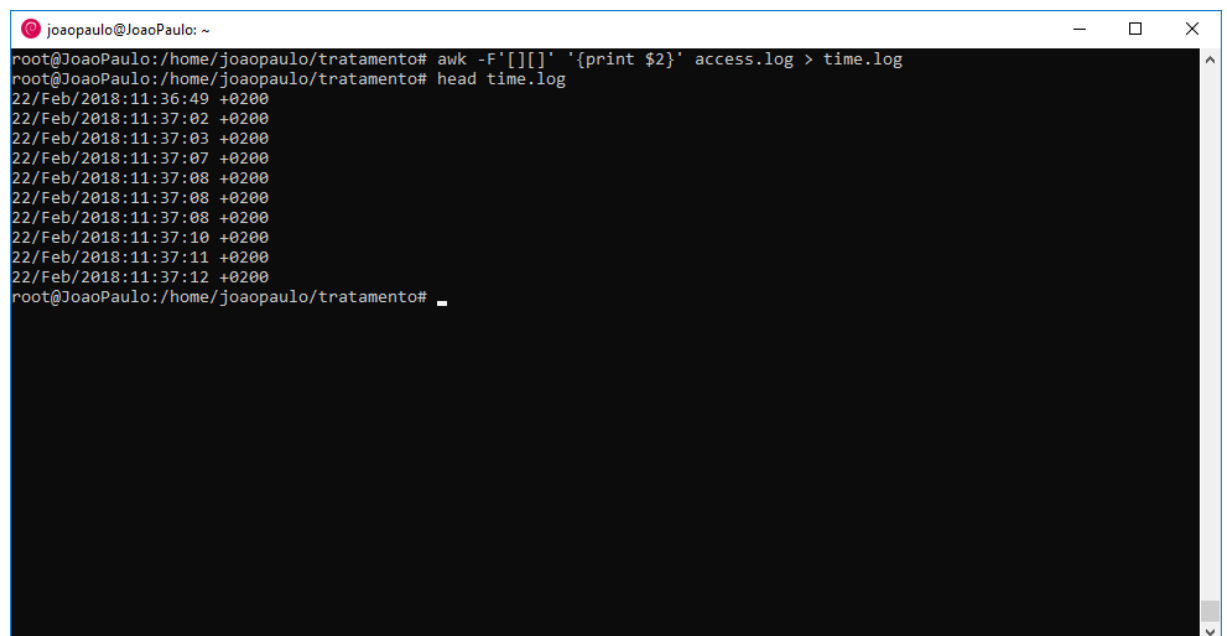
```
joaopaulo@JoaoPaulo: ~  
root@JoaoPaulo:/home/joaopaulo/tratamento# head access.log  
108.62.115.226 - - [22/Feb/2018:11:36:49 +0200] "GET / HTTP/1.1" 403 1069 "-" "Pingdom.com_bot_version_1.4_(http://www.p  
ingdom.com/)"  
77.139.51.223 - - [22/Feb/2018:11:37:02 +0200] "GET / HTTP/1.1" 200 31958 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)  
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36"  
77.139.51.223 - - [22/Feb/2018:11:37:03 +0200] "GET / HTTP/1.1" 200 31958 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)  
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36"  
77.139.51.223 - - [22/Feb/2018:11:37:07 +0200] "GET /login/index.php HTTP/1.1" 303 446 "http://www.devqa71.robotec.co.il  
/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36"  
77.139.51.223 - - [22/Feb/2018:11:37:08 +0200] "GET /login/index.php HTTP/1.1" 200 23080 "http://www.devqa71.robotec.co.  
il/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.3  
6"  
77.139.51.223 - - [22/Feb/2018:11:37:08 +0200] "GET /theme/image.php/intelitek/core/1453127471/i/warning HTTP/1.1" 200 9  
74 "http://www.devqa71.robotec.co.il/login/index.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHT  
ML, like Gecko) Chrome/63.0.3239.132 Safari/537.36"  
77.139.51.223 - - [22/Feb/2018:11:37:08 +0200] "GET /theme/image.php/intelitek/core/1453127471/help HTTP/1.1" 200 766 "h  
tp://www.devqa71.robotec.co.il/login/index.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, l  
ike Gecko) Chrome/63.0.3239.132 Safari/537.36"  
77.139.51.223 - - [22/Feb/2018:11:37:10 +0200] "POST /login/index.php HTTP/1.1" 303 460 "http://www.devqa71.robotec.co.i  
l/login/index.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.13  
2 Safari/537.36"  
77.139.51.223 - - [22/Feb/2018:11:37:11 +0200] "GET /login/index.php?testsession=2 HTTP/1.1" 303 431 "http://www.devqa71  
.robotec.co.il/login/index.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome  
/63.0.3239.132 Safari/537.36"  
77.139.51.223 - - [22/Feb/2018:11:37:12 +0200] "GET / HTTP/1.1" 200 52618 "http://www.devqa71.robotec.co.il/login/index.  
php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.3  
6"  
root@JoaoPaulo:/home/joaopaulo/tratamento#
```

2º - Confirmado o padrão do arquivo, o campo do IP será extraído e enviado para um arquivo separado chamado “ip.log”.



```
joaopaulo@JoaoPaulo: ~  
root@JoaoPaulo:/home/joaopaulo/tratamento# awk '{print $1}' access.log > ip.log  
root@JoaoPaulo:/home/joaopaulo/tratamento# head ip.log  
108.62.115.226  
77.139.51.223  
77.139.51.223  
77.139.51.223  
77.139.51.223  
77.139.51.223  
77.139.51.223  
77.139.51.223  
77.139.51.223  
77.139.51.223  
root@JoaoPaulo:/home/joaopaulo/tratamento#
```

3º - Posteriormente, o campo da data será extraído e enviado também para outro arquivo.



```
joaopaulo@JoaoPaulo: ~  
root@JoaoPaulo:/home/joaopaulo/tratamento# awk -F'[ ]' '{print $2}' access.log > time.log  
root@JoaoPaulo:/home/joaopaulo/tratamento# head time.log  
22/Feb/2018:11:36:49 +0200  
22/Feb/2018:11:37:02 +0200  
22/Feb/2018:11:37:03 +0200  
22/Feb/2018:11:37:07 +0200  
22/Feb/2018:11:37:08 +0200  
22/Feb/2018:11:37:08 +0200  
22/Feb/2018:11:37:08 +0200  
22/Feb/2018:11:37:10 +0200  
22/Feb/2018:11:37:11 +0200  
22/Feb/2018:11:37:12 +0200  
root@JoaoPaulo:/home/joaopaulo/tratamento#
```

4º - Para extrair o campo do browser, em primeiro lugar, deve-se conferir o campo em que se encontram tais informações. O comando “awk -F\" '{ print \$1 }' access.log | head” deve ser executado e o valor de \$1 deve ser incrementado até que a saída

seja satisfatória. No caso do exemplo, o valor encontrado foi \$6. Feito isso, a saída do comando será enviada para um arquivo separado, de maneira semelhante aos outros campos.

```
joaopaulo@JoaoPaulo: ~  
root@JoaoPaulo:/home/joaopaulo/tratamento# awk -F\" '{ print $6 }' access.log | head  
Pingdom.com_bot_version_1.4_(http://www.pingdom.com/)  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
root@JoaoPaulo:/home/joaopaulo/tratamento#
```

```
joaopaulo@JoaoPaulo: ~  
root@JoaoPaulo:/home/joaopaulo/tratamento# awk -F\" '{ print $6 }' access.log > browser.log  
root@JoaoPaulo:/home/joaopaulo/tratamento# head browser.log  
Pingdom.com_bot_version_1.4_(http://www.pingdom.com/)  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36  
root@JoaoPaulo:/home/joaopaulo/tratamento#
```

5° - Com os três arquivos, deve-se uni-los em um só, definindo o caractere separador que o software requisita, no caso o "&". O comando "paste -d& ip.log time.log browser.log > access\_mofif.log" foi utilizado. Por último, o comando "head" foi usado para verificar a versão final do arquivo log.



```
joaopaulo@JoaoPaulo: ~
root@JoaoPaulo:/home/joaopaulo/tratamento# paste -d\& ip.log time.log browser.log > access_modif.log
root@JoaoPaulo:/home/joaopaulo/tratamento# head access_modif.log
108.62.115.226&22/Feb/2018:11:36:49 +0200&Pingdom.com_bot_version_1.4_(http://www.pingdom.com/)
77.139.51.223&22/Feb/2018:11:37:02 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
77.139.51.223&22/Feb/2018:11:37:03 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
77.139.51.223&22/Feb/2018:11:37:07 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
77.139.51.223&22/Feb/2018:11:37:08 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
77.139.51.223&22/Feb/2018:11:37:08 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
77.139.51.223&22/Feb/2018:11:37:10 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
77.139.51.223&22/Feb/2018:11:37:11 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
77.139.51.223&22/Feb/2018:11:37:12 +0200&Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
) Chrome/63.0.3239.132 Safari/537.36
root@JoaoPaulo:/home/joaopaulo/tratamento#
```

## 5 – Conclusão

Arquivos log são fontes valiosas de informações usadas por desenvolvedores e gestores. Diversos aplicativos e sistemas geram e mantêm tais arquivos que, se não forem bem geridos, podem causar sérios problemas.

No foco do trabalho, está o arquivo log de acesso do servidor http. Dele, pode-se extrair diversas informações úteis, como o horário de pico de acesso de determinada página, quantos acessos por dia ou, até mesmo, a localização geográfica aproximada dos clients, com base nestes dados, conseguiremos fazer análises estatísticas e gerar informações para tomada de decisão gerencial.

## 6 – Referência bibliográfica

STRONG SECURITY (Brasil). **Você sabe o que é log de dados?:** Entenda sua importância. 2017. Disponível em: <<https://www.strongsecurity.com.br/voce-sabe-o-que-e-log-de-dados-entenda-sua-importancia/>>. Acesso em: 30 nov. 2018.

LOG Files. 2018. Disponível em: <<https://httpd.apache.org/docs/1.3/logs.html>>. Acesso em: 30 nov. 2018.