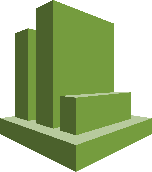
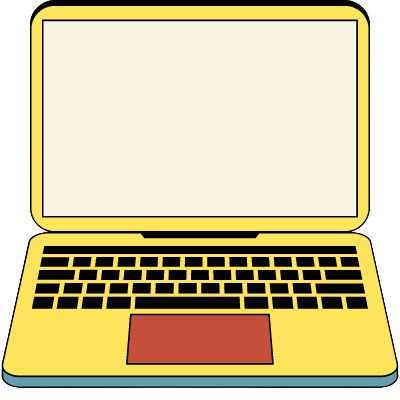
Habilitar Cloudwatch

**Christian Millán Soria**





- Curso: 2023-2024

- Clase: 2º DAW Tarde

- Profesor: David Hormigo Ramírez

- Módulo: Despliegue de Aplicaciones Web

- Fecha: 05/12/2023

**1. Perfil de instancia de IAM** 3

**2. Cliente de Cloudwatch** 4

**2.1. Sistema operativo** 4

**2.10. Por cada núcleo de CPU** 7

**2.11. Dimensiones de EC2** 7

**2.12. Agregar dimensiones de EC2** 7

**2.13. Métricas cada minuto** 7

**2.14. Tipo de métricas** 7

**2.15. ¿Satisfecho?** 8

**2.16. Cloudwatch previo** 8

**2.17. Fichero externo** 8

**2.18. Ruta de lectura** 8

**2.19. Nombre del grupo de errores** 9

**2.2. Región predeterminada para los metadatos** 5

**2.20. Nombre de la rama** 9

**2.21. Días de retención** 9

**2.22. ¿Otros archivos?** 9

**2.23. Ruta nueva** 10

**2.24. Nombre del segundo grupo** 10

**2.25. Nombre de la segunda rama** 10

**2.26. Días de retención** 10

**2.27. Último archivo** 11

**2.28. X-Ray** 11

**2.29. Almacenar el archivo de configuración en SSM** 11

**2.3. Usuario predeterminado** 5

**2.4. StatsD Daemon** 5

**2.5. Puerto** 5

**2.6. Intervalo** 6

**2.7. Intervalo de métricas** 6

**2.8. Monitoreo de métricas de CollectD** 6

**2.9. Métricas por hardware** 6

**3. Formato de los logs** 12

**4. Comprobación** 13

Para saber qué está pasando en nuestro servidor en todo momento o, en caso de ocurrir un error inesperado, poder solventarlo de la forma más eficaz posible, necesitamos hacer uso de los logs que este nos proporciona.

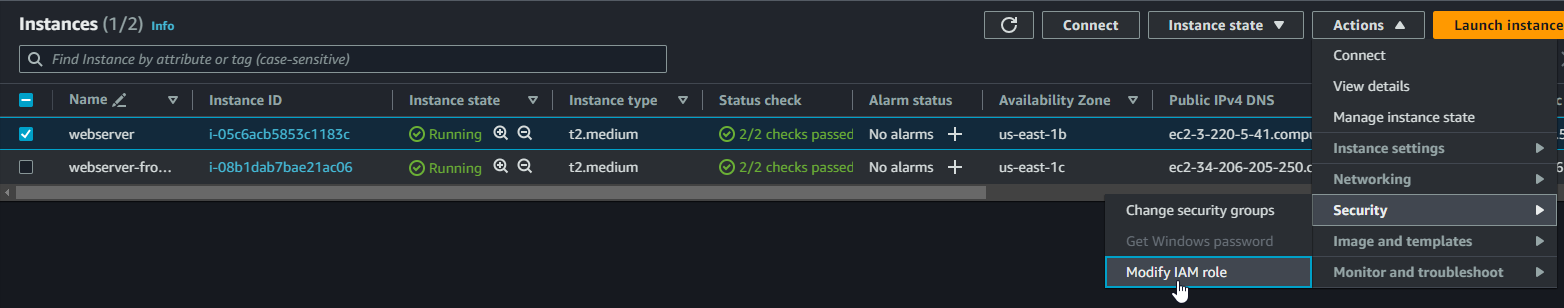
Para realizar esto de una manera más concentrada y rápida, AWS creó un servicio llamado [“**Cloudwatch**”🔗](https://aws.amazon.com/cloudwatch/), con el cual podemos crear archivos de logs personalizados para nuestras máquinas y así poder clasificar los mismos para poder ver qué pasa constantemente en nuestro servidor.

**1. Perfil de instancia de IAM**

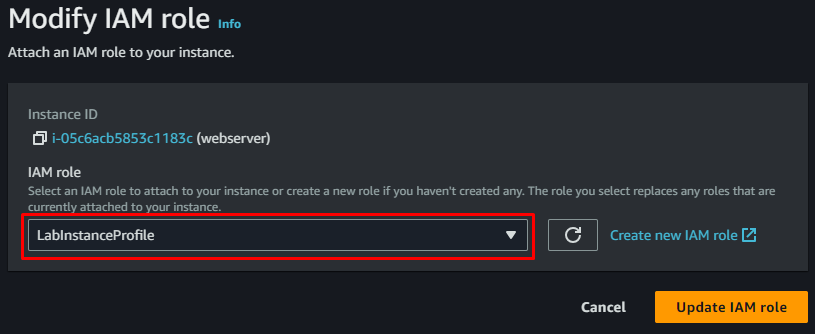
El primer paso a seguir es establecer un [**rol de IAM para nuestras instancias🔗**](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html), es decir, una “identidad” para nuestra instancia para que pueda acceder a ciertos servicios y cuente con unos permisos determinados.

Nos dirigimos al listado de instancias de nuestra cuenta de AWS, seleccionamos una de las dos instancias con las que contamos y hacemos clic en el desplegable de **Actions**.

Entramos en la subopción **Security** y hacemos clic en la opción **Modify IAM role**.



Una vez dentro, en el desplegable que se nos proporciona seleccionamos la opción **LabInstanceProfile**.



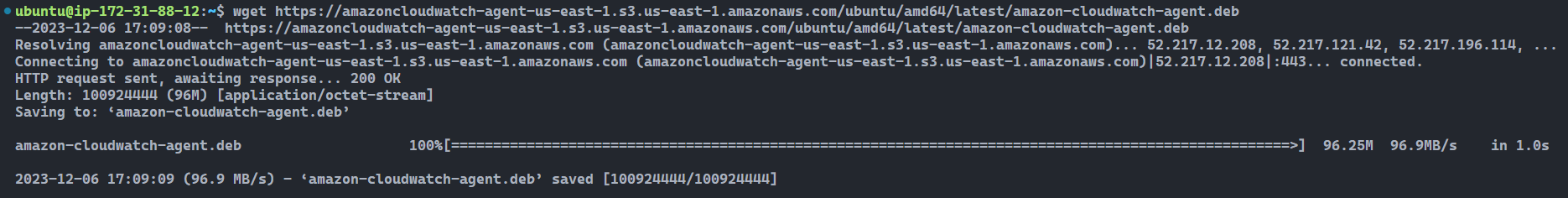
Guardamos los cambios y hacemos lo mismo con la otra instancia de nuestra lista.

**2. Cliente de Cloudwatch**

Lo siguiente es instalar el cliente de Cloudwatch en una de nuestras máquinas. Nos conectamos mediante SSH y ejecutamos el siguiente comando:

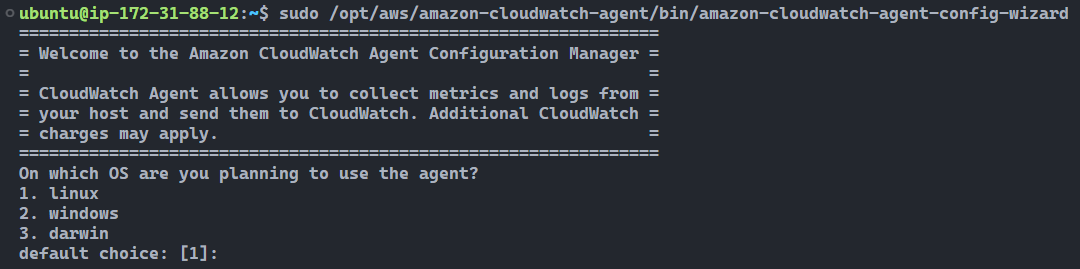
**wget** [**https://amazoncloudwatch-agent-us-east-1.s3.us-east-1.amazonaws.com/ubuntu/amd64/latest/amazon-cloudwatch-agent.deb**](https://amazoncloudwatch-agent-us-east-1.s3.us-east-1.amazonaws.com/ubuntu/amd64/latest/amazon-cloudwatch-agent.deb)

Se descarga el cliente.



El siguiente comando a ejecutar es el comando [**sudo dpkg -i -E ./amazon-cloudwatch-agent.deb**🔗.](https://explainshell.com/explain?cmd=sudo+dpkg+-i+-E+.%2Famazon-cloudwatch-agent.deb)

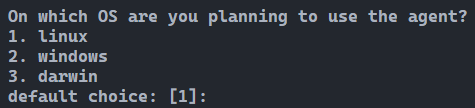
Y por último el comando **sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard**, el cual ejecuta la configuración del nuevo archivo ce configuración de Cloudwatch.



Se nos muestra una lista de apartados que debemos ir rellenando paso a paso.

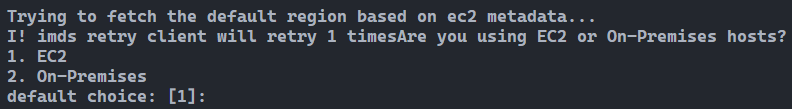
**2.1. Sistema operativo**

Seleccionamos la opción **1**, ya que instalaremos el agente de Cloudwatch en un sistema **Linux**.



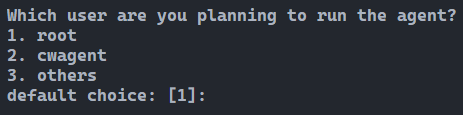
**2.2. Región predeterminada para los metadatos**

Elegimos la opción de **EC2**, la **1**, ya que es cómo estamos trabajando con AWS.



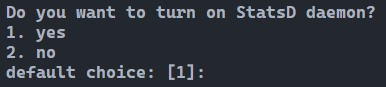
**2.3. Usuario predeterminado**

Para utilizar el usuario **root**, dejamos la opción **1** como la seleccionada.



**2.4. StatsD Daemon**

**Habilitamos el Daemon**.



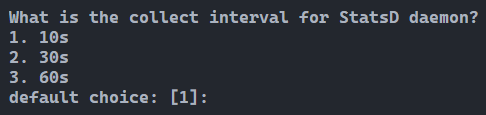
**2.5. Puerto**

Utilizamos el **puerto predeterminado** para no interferir con posibles configuraciones futuras o anteriores.



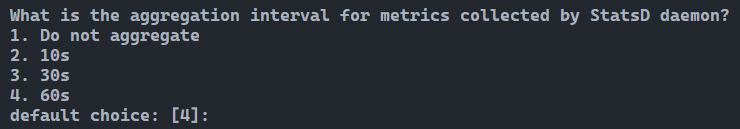
**2.6. Intervalo**

Dejamos el intervalo con su **valor predeterminado**.



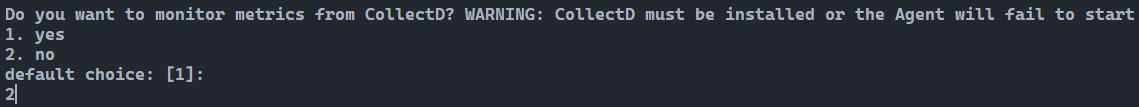
**2.7. Intervalo de métricas**

Establecemos que el intervalo para la recolecta de métricas por parte del Daemon se haga cada **60 segundos**.



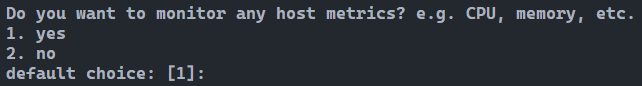
**2.8. Monitoreo de métricas de CollectD**

**No** queremos llevar a cabo el monitoreo de métricas, ya que requiere de la instalación de un software adicional.



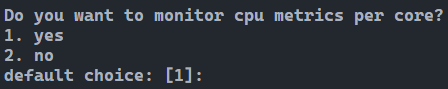
**2.9. Métricas por hardware**

Elegimos **la opción que nos permite monitorear** las métricas de la CPU, memoria, etc.



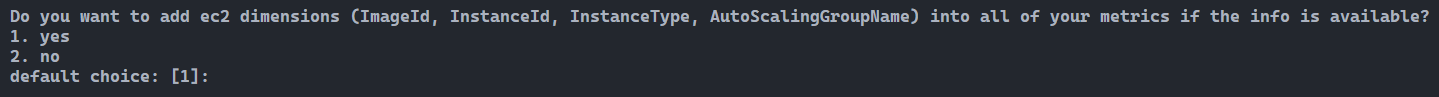
**2.10. Por cada núcleo de CPU**

**Sí** queremos monitorear las métricas separadas por núcleos.



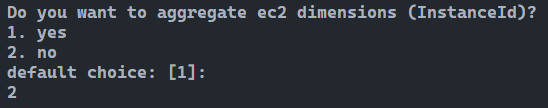
**2.11. Dimensiones de EC2**

**Habilitamos** las dimensiones de EC2 para poder contar con datos exactos de la instancia si están disponibles.



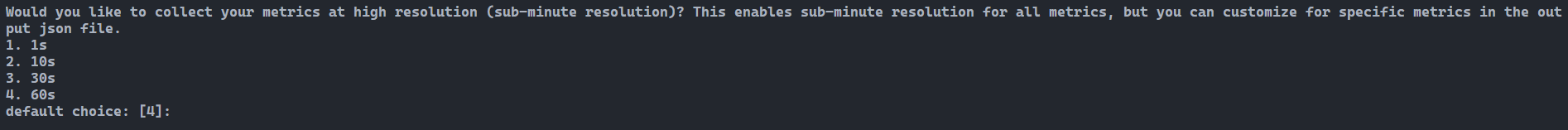
**2.12. Agregar dimensiones de EC2**

Escogemos la opción **No**.



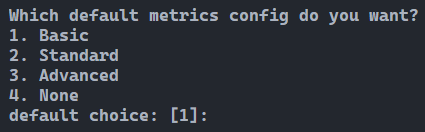
**2.13. Métricas cada minuto**

Escogemos que las métricas se guarden cada **60 segundos**.



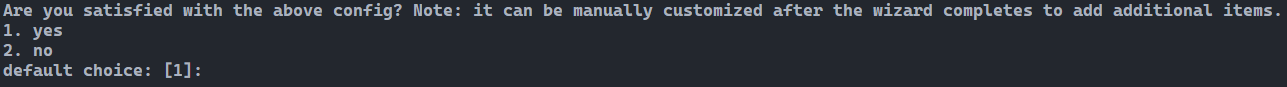
**2.14. Tipo de métricas**

Escogemos el tipo **Basic** para las métricas que recopilaremos.



**2.15. ¿Satisfecho?**

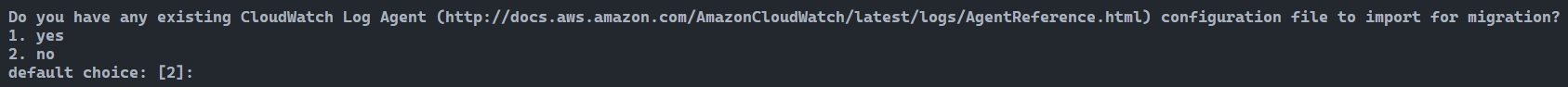
**Sí**.



Se ha generado el archivo de configuración.

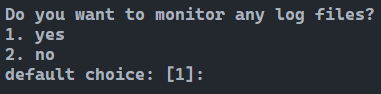
**2.16. Cloudwatch previo**

**No**, no tenemos un agente previo del que queremos migrar.



**2.17. Fichero externo**

Escogemos la **primera opción** para poder especificar el archivo donde guardaremos los logs.



**2.18. Ruta de lectura**

Establecemos **la ruta para que Cloudwatch lea desde la misma**.



**2.19. Nombre del grupo de errores**

**Establecemos un nombre para el grupo**.



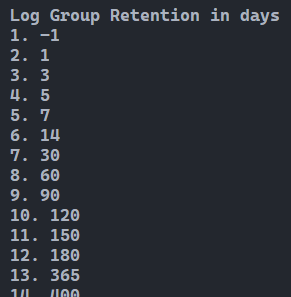
**2.20. Nombre de la rama**

Establecemos un nombre para la rama. Lo dejamos de **forma predeterminada**.



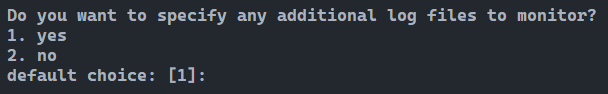
**2.21. Días de retención**

Dejamos los días antes de la eliminación automática de los logs de **forma predeterminada**.



**2.22. ¿Otros archivos?**

Seleccionamos la **primera opción** para poder agregar otra ruta diferente.



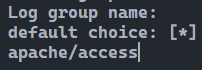
**2.23. Ruta nueva**

Establecemos la **segunda ruta**.



**2.24. Nombre del segundo grupo**

Escribimos el **nombre para el segundo grupo de logs**.



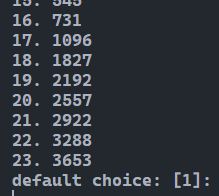
**2.25. Nombre de la segunda rama**

Hacemos lo mismo que con la primera, lo dejamos en su **valor predeterminado**.



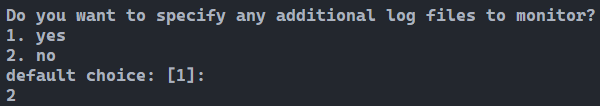
**2.26. Días de retención**

Dejamos los días **como en el caso anterior**.



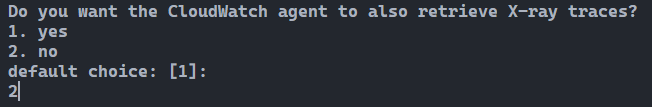
**2.27. Último archivo**

**No** creamos ningún archivo nuevo.



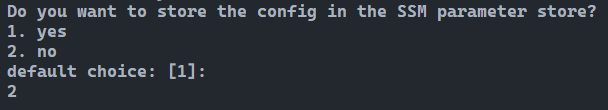
**2.28. X-Ray**

**No** habilitamos las trazas de “rayos x”.



**2.29. Almacenar el archivo de configuración en SSM**

**No** almacenamos la configuración.



Hemos terminado este proceso.

**3. Formato de los logs**

Vamos a editar el archivo de configuración del sitio que tenemos activo en nuestra máquina.

Dentro de la directiva de virtual host añadimos las rutas para los archivos que almacenarán nuestros logs y establecemos el formato de los mismos al final de cada línea.



El formato “**cloudwatch**” no está creado todavía, pero es lo siguiente que debemos hacer.

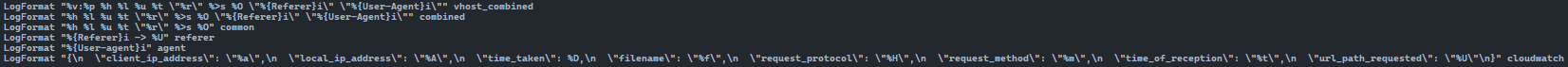
Entramos a editar el archivo **/etc/apache2/apache2.conf**.

Buscamos la palabra **logformat**. Nos encontramos con una lista de formatos ya creados de forma predeterminada por Apache para los logs predeterminados del servidor.

Creamos una nueva línea que comience con la palabra clave, con el nombre del formato al final y creamos un formato personalizado.

\* [**Lista de los posibles elementos aquí**🔗](https://httpd.apache.org/docs/2.4/mod/mod_log_config.html).

**LogFormat "{\n \"client\_ip\_address\": \"%a\",\n \"local\_ip\_address\": \"%A\",\n \"time\_taken\": %D,\n \"filename\": \"%f\",\n \"request\_protocol\": \"%H\",\n \"request\_method\": \"%m\",\n \"time\_of\_reception\": \"%t\",\n \"url\_path\_requested\": \"%U\"\n}" cloudwatch**



Ejecutamos el siguiente comando:

**cd /var/log/apache2/ && sudo mkdir access error && sudo touch access/access.log error/error.log**

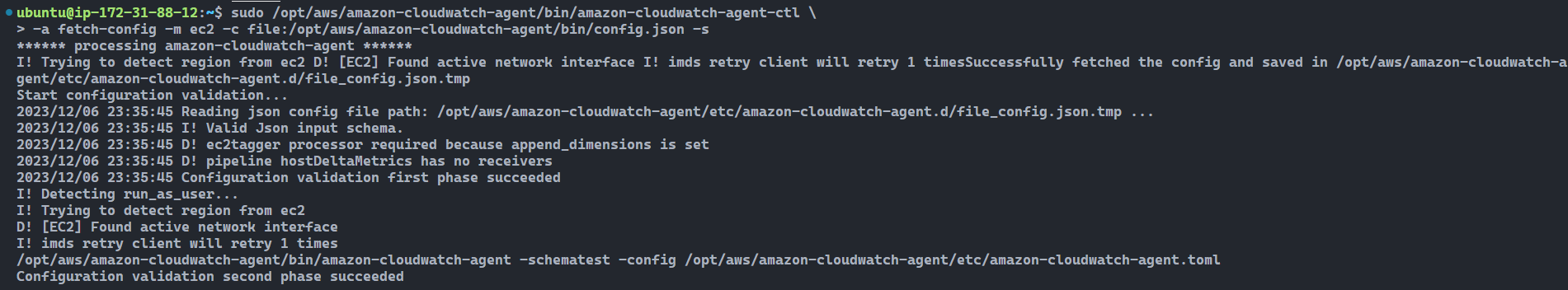
Reiniciamos Apache.

**4. Comprobación**

Para acabar, utilizaremos el comando **sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl \** para activar la escucha desde el archivo de configuración principal de Cloudwatch.

Se nos abre un brazo para escribir otra línea, donde ingresaremos lo siguiente:

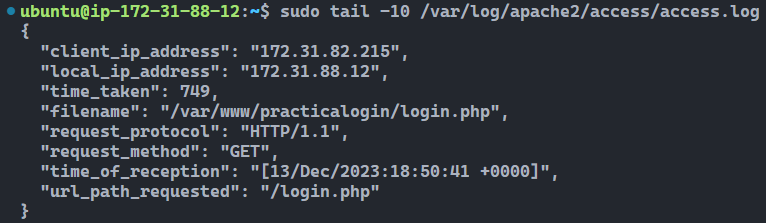
**-a fetch-config -m ec2 -c file:/opt/aws/amazon-cloudwatch-agent/bin/config.json -s**



Una vez hemos hecho todo esto, podemos pasar a reiniciar el servicio y habilitarlo para que se encienda automáticamente.



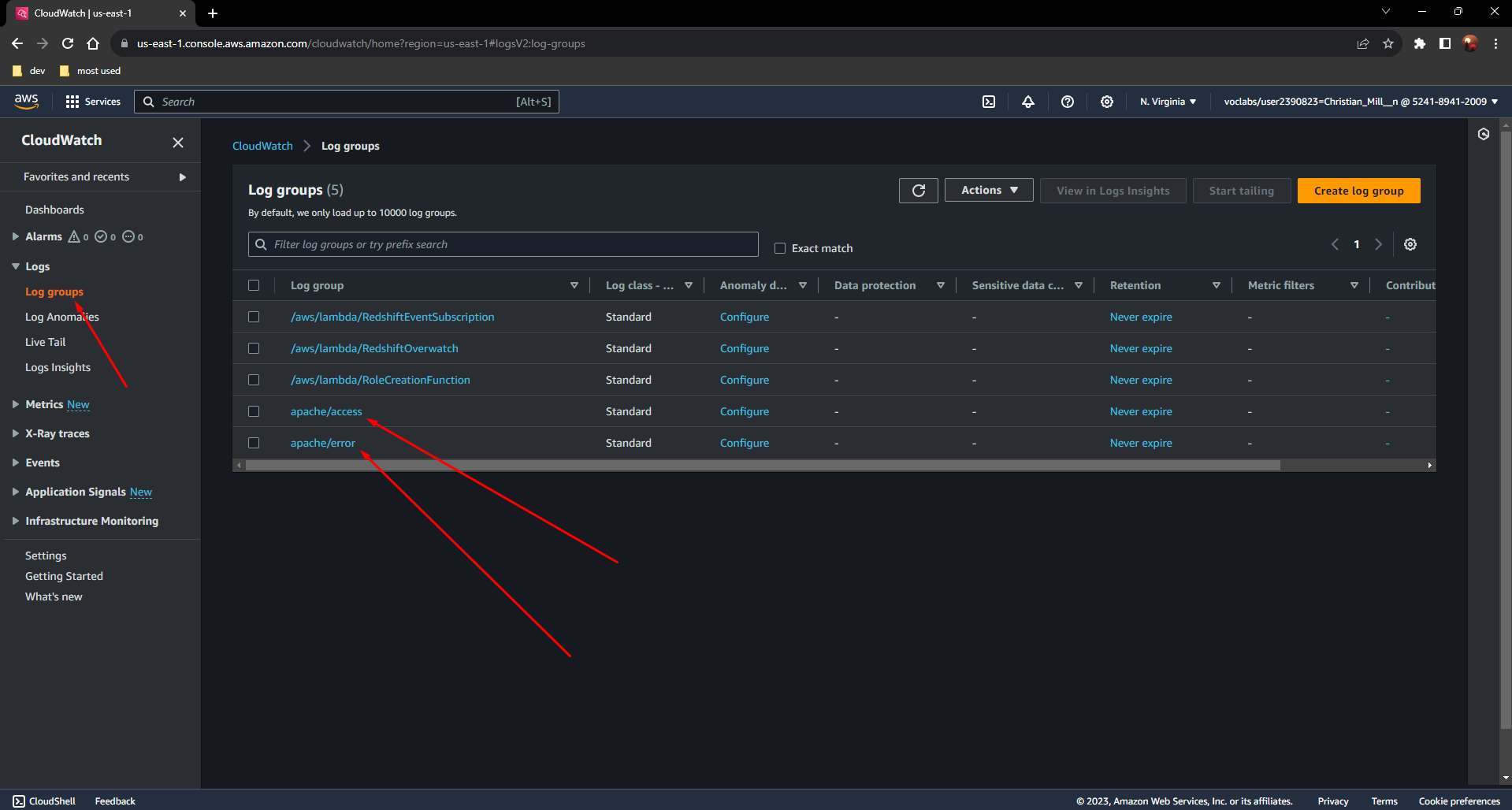
Para llevar a cabo la comprobación, podemos entrar a la web de nuestro servidor desde el navegador web, después volvernos al terminal de comandos de nuevo y ejecutar el comando **sudo tail -10 /var/log/apache2/access/access.log**, con el cual podremos ver las últimas 10 líneas del archivo de logs donde se están almacenando los logs de acceso.



Adicionalmente, podemos modificar el código de la página para escribir algo no funcional y se generará un error, el cual se podrá ver en el archivo correspondiente.



\* Estos logs pueden verse de forma más visual en el apartado de **Log Groups** en el servicio de **Cloudwatch en AWS**.



\* También podemos ejecutar una query con un limit de 20 (lo que corresponde a los últimos 20 logs almacenados) seleccionando el grupo deseado en el desplegable y podemos ver lo mismo.

