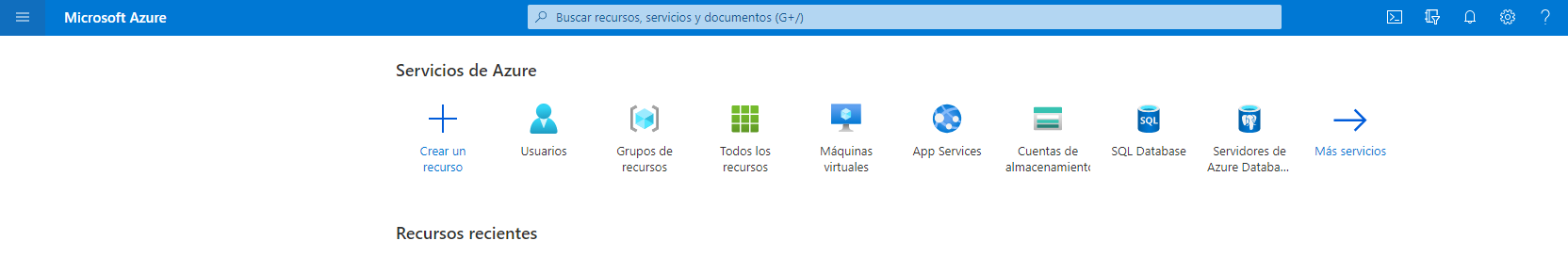
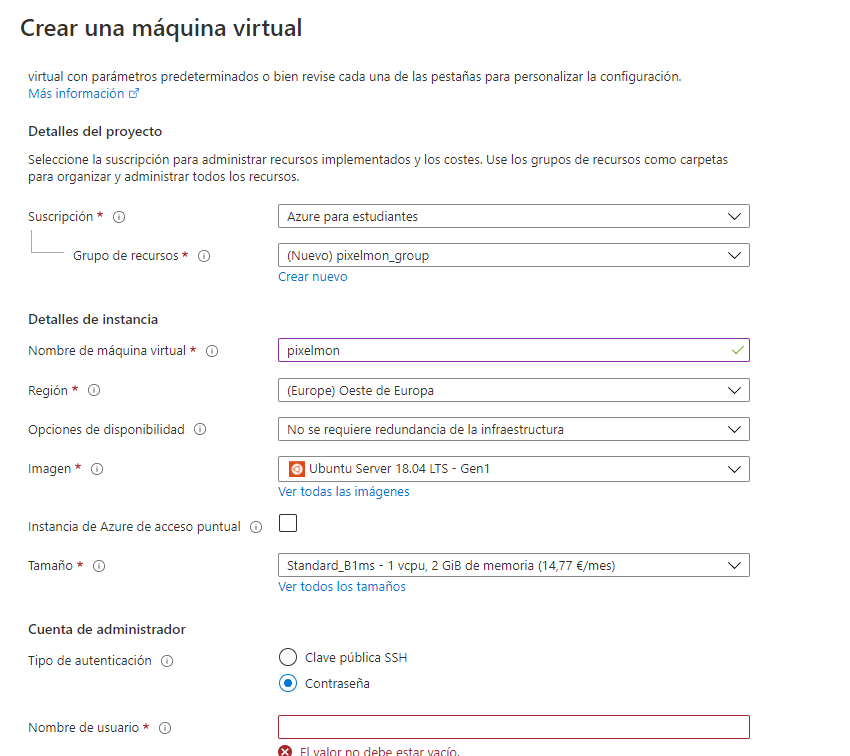
Paso \*: Crear una cuenta de Azure (puedes solicitar cuenta de estudiantes con 100$ gratuitos)

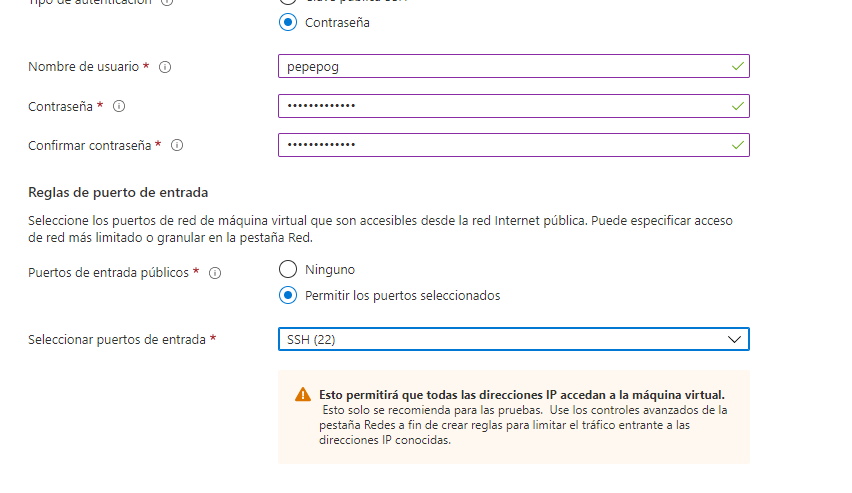
Paso 1: Una vez creada la cuenta, vamos a crear nuestra máquina servidor en **Crear Recurso:**



Paso 2: Seleccionamos Ubuntu 18.04 con 2G RAM (nosotros hemos utilizado estos recursos puesto que lo vemos apropiado).

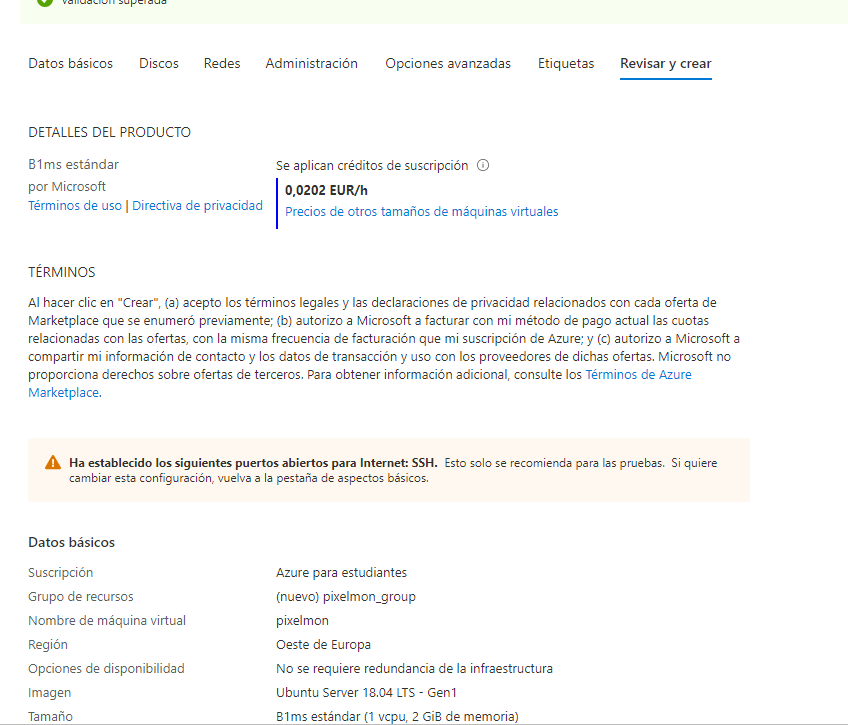


Paso 3: Vamos a definir un user/pass para la identificación por SSH hacia nuestra máquina:



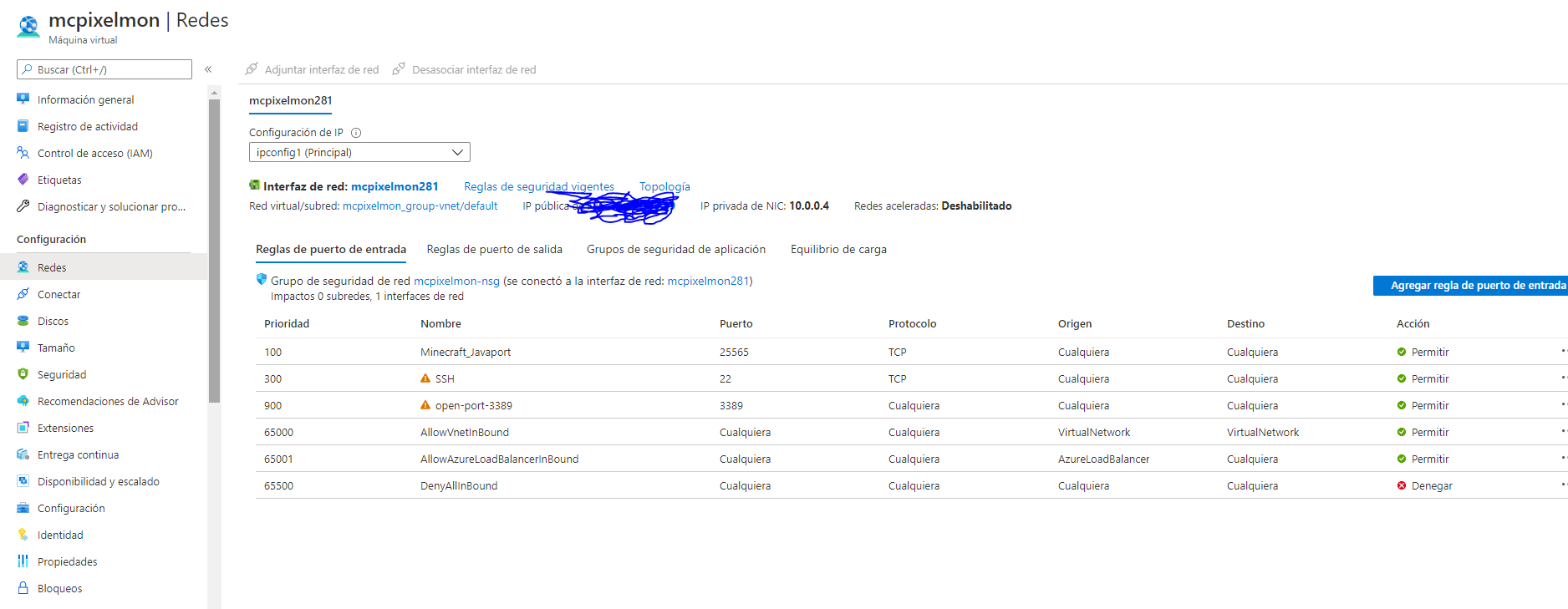
\*Los puertos los abriremos más adelante, de momento abrir solo SSH para la configuración del servidor.

Paso 4: Revisar y Crear:



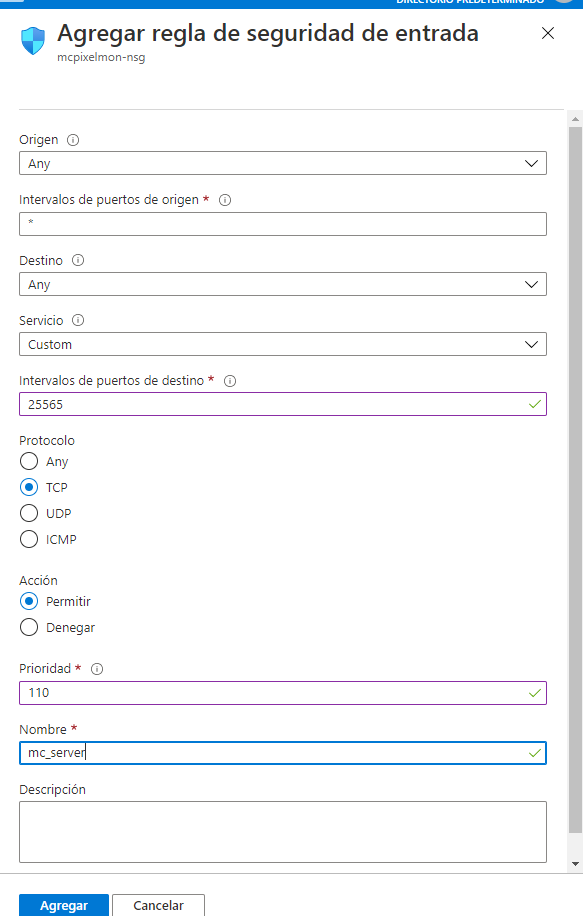
Paso 5: una vez creada la máquina vamos a definir una serie de reglas de conexión:

Vamos a nuestra máquina recientemente creada, redes:



Pulsamos en el recuadro azul para establecer una regla de entrada hacia el puerto 25565 (es el que utiliza mc por defecto, puede ser el que queráis siempre que configuréis tanto en Linux como en Azure.

Regla de entrada (copiar/pegar):



Hasta aquí llega toda la configuración previa de Azure, ahora ya tenemos nuestro servidor Ubuntu 18.04 listo para configurar:

Paso 1: Acceso a nuestro servidor:

Para acceder a vuestra máquina virtual debéis ingresar el siguiente comando en vuestra CMD:

ssh USUARIO\_SSH@IP\_SERVIDOR

**\*NOTA: el usuario es el que hemos definido anteriormente en el paso 3 de AZURE.**

Paso 2: Instalación de openjdk:

Necesitamos java para desplegar nuestro servidor, por lo que vamos a ingresar los siguientes comandos:

sudo add-apt-repository ppa:openjdk-r/ppa

sudo apt-get update

sudo apt-get install openjdk-8-jdk

Paso 3: Crear una carpeta para el servidor:

Creamos una carpeta en el /home/user con X nombre:



Paso 4: Descarga de Forge + subida al servidor:

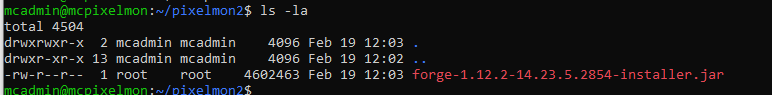
Vamos al siguiente LINK de forge y descargamos la versión 1.12.2 (la última para Pixelmon)

<https://files.minecraftforge.net/maven/net/minecraftforge/forge/index_1.12.2.html>

Una vez descargado, vamos a subir el archivo .jar hacia nuestro servidor con el siguiente comando (IMPORTANTE: hacer este comando fuera de nuestra máquina virtual, abrid un nuevo CMD):

scp /directorio\_local/ ARCHIVO\_FORGE usuario\_remoto@host\_remoto:/algun\_directorio\_remoto

NOTA: Podéis hacerlo también por FTP, VSFTP etc, lo importante es subir el archivo de FORGE en la carpeta que hemos creado en el servidor.



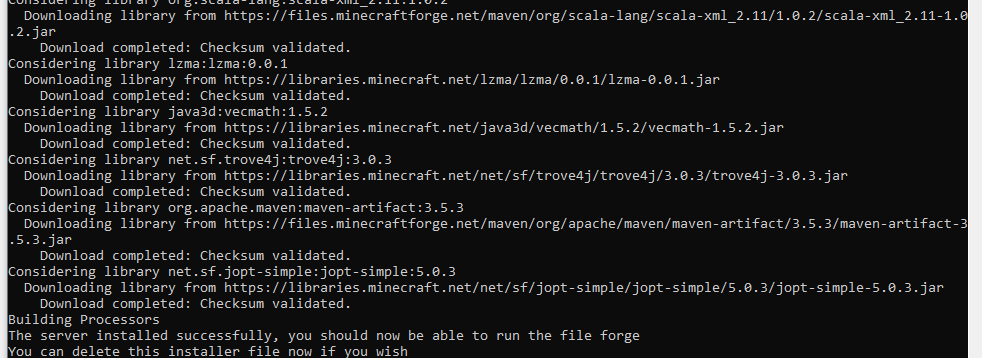
Paso 5: instalación de Forge en el servidor:

Ahora vamos a instalar Forge + servidor de minecraft en nuestra MV, por lo que vamos a ejecutar sobre la carpeta del servidor, el siguiente comando:

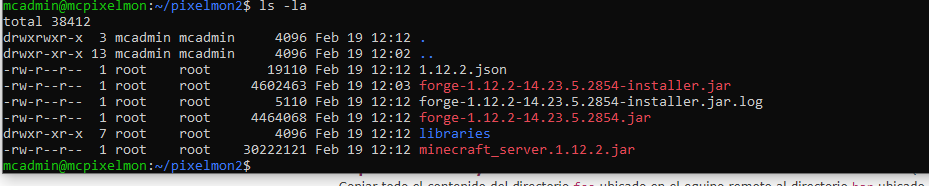


sudo java -jar forge-1.12.2-14.23.5.2854-installer.jar –installServer

Una vez realizado este comando, vemos que funcione:



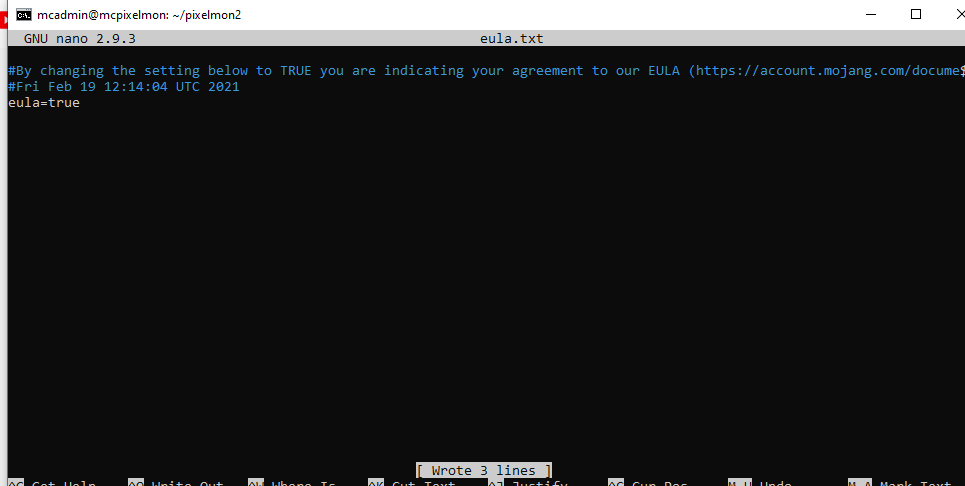
Podemos ver nuevos archivos en la carpeta:



Ahora ejecutamos el nuevo archivo de forge con el mismo comando que antes (prácticamente, revisad sintaxis)



Debemos activar el archivo eula.txt, simplemente editamos con sudo nano eula.txt y cambiamos a “true” el parámetro:



Ahora vamos a re-ejecutar el anterior comando:



Una vez terminado, tenemos un nuevo archivo llamado server.properties, lo editamos con sudo nano.

Parámetros importantes:

 Cambiar a “false” si usuarios no premium quieren jugar

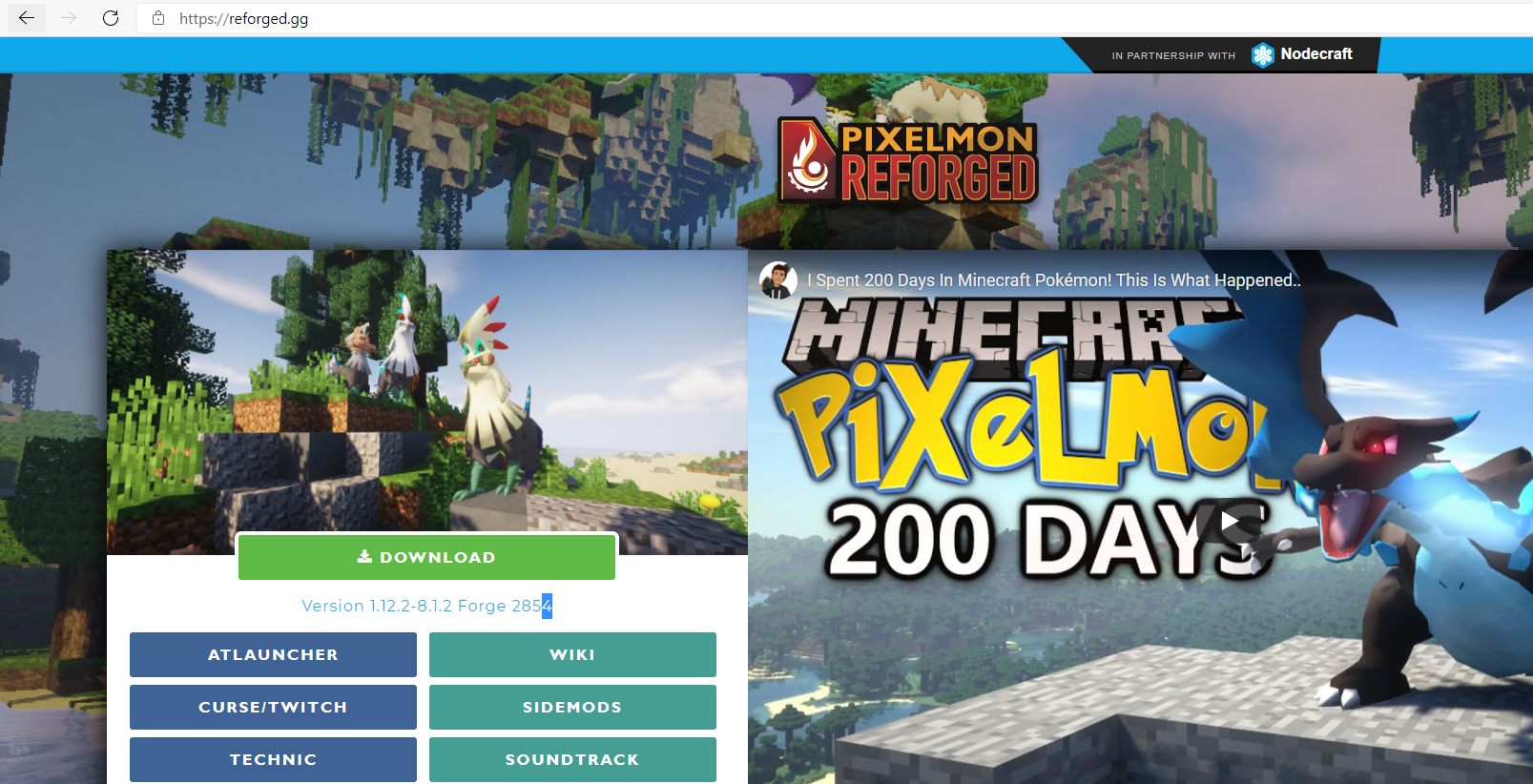
 IP del servidor (para verla, usar ifconfig, será algo así como 10.0.0.4)

En un futuro, aquí podéis modificar todas las configuraciones del servidor.

Paso 5: Instalación del mod Pixelmon

Esto es lo más sencillo, vamos al la web de pixelmon y nos descargamos la versión para la 1.12.2:

<https://reforged.gg/>



Una vez descargado, podemos enviar el archivo con el comando SCP anteriormente usado o con FTP, como veáis.

Una vez dentro del servidor, debemos meter el archivo pixelmon dentro de la carpeta mods:



Paso 6: Ejecutar servidor

Para ejecutar el servidor debemos usar el siguiente comando:

sudo java -Xmx<MaxRAM> -Xms<MinRAM> -jar <FileForge>.jar



Una vez ejecutado, podremos conectarnos al servidor desde nuestro Launcher, con la IP de azure:25565

NOTA: se puede usar la herramienta **screen** para ejecutar el servidor en segundo plano.