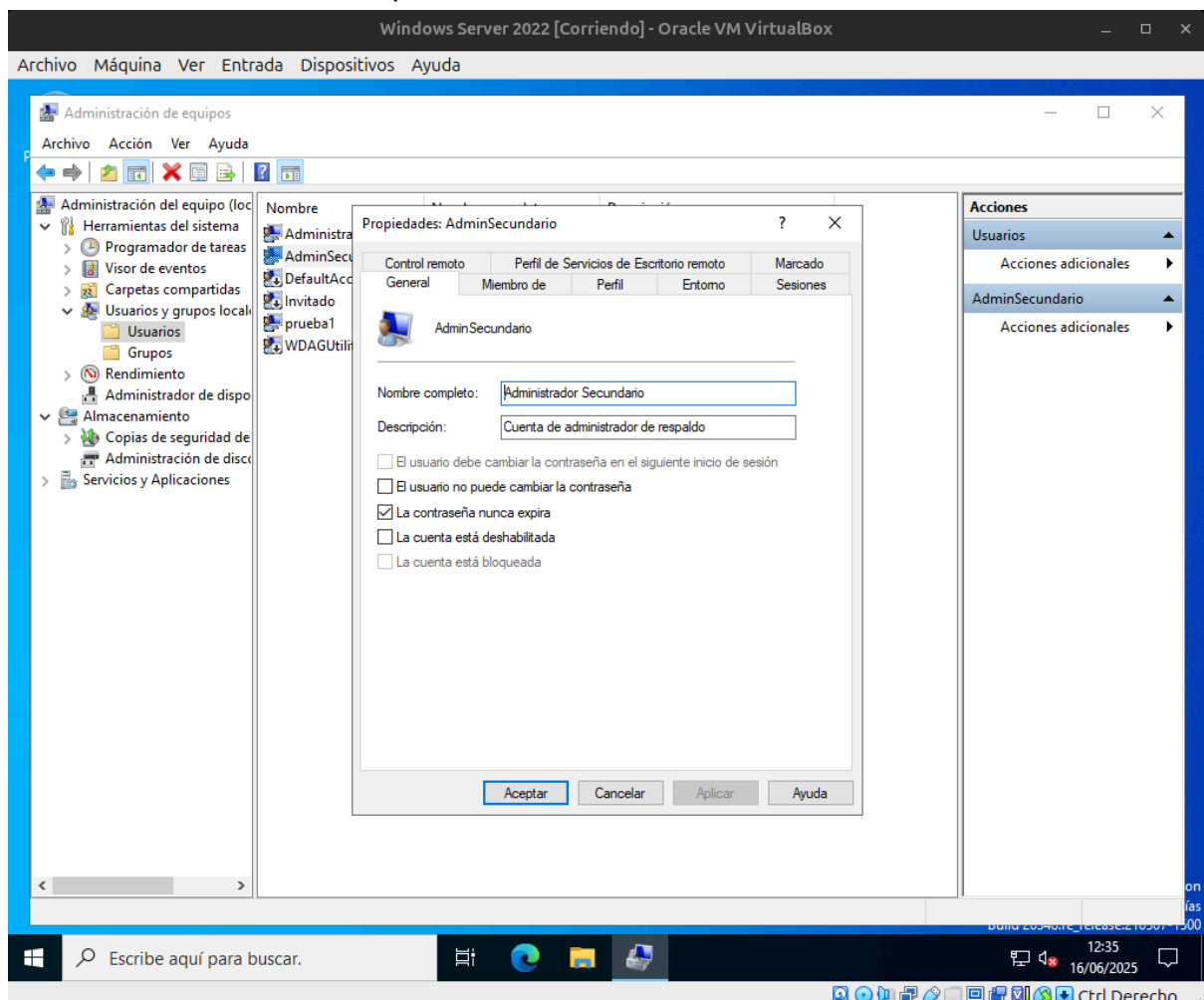
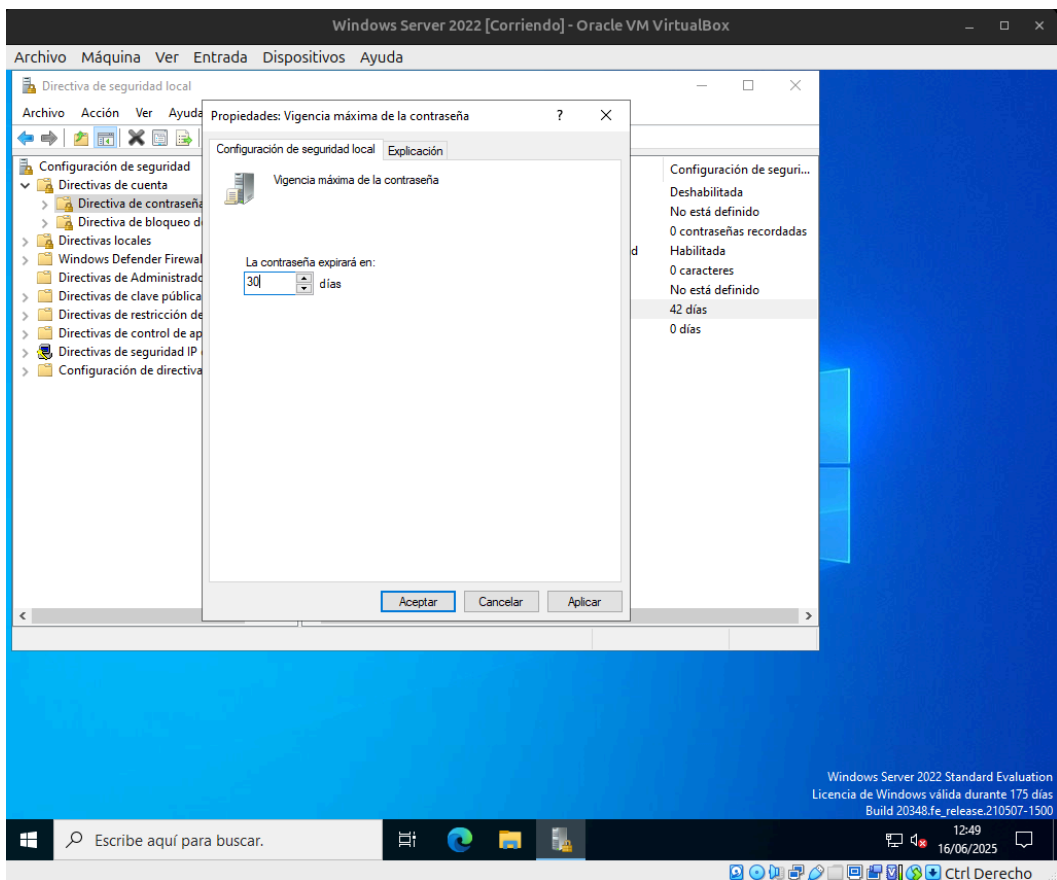
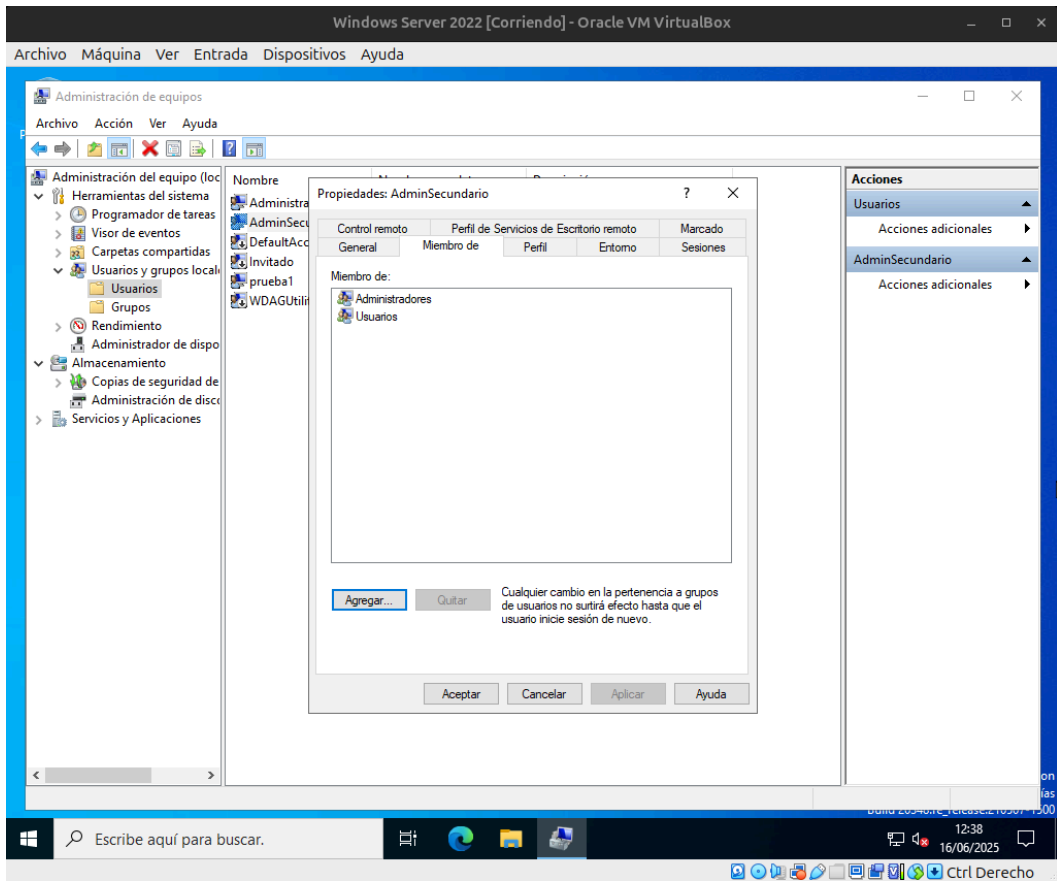
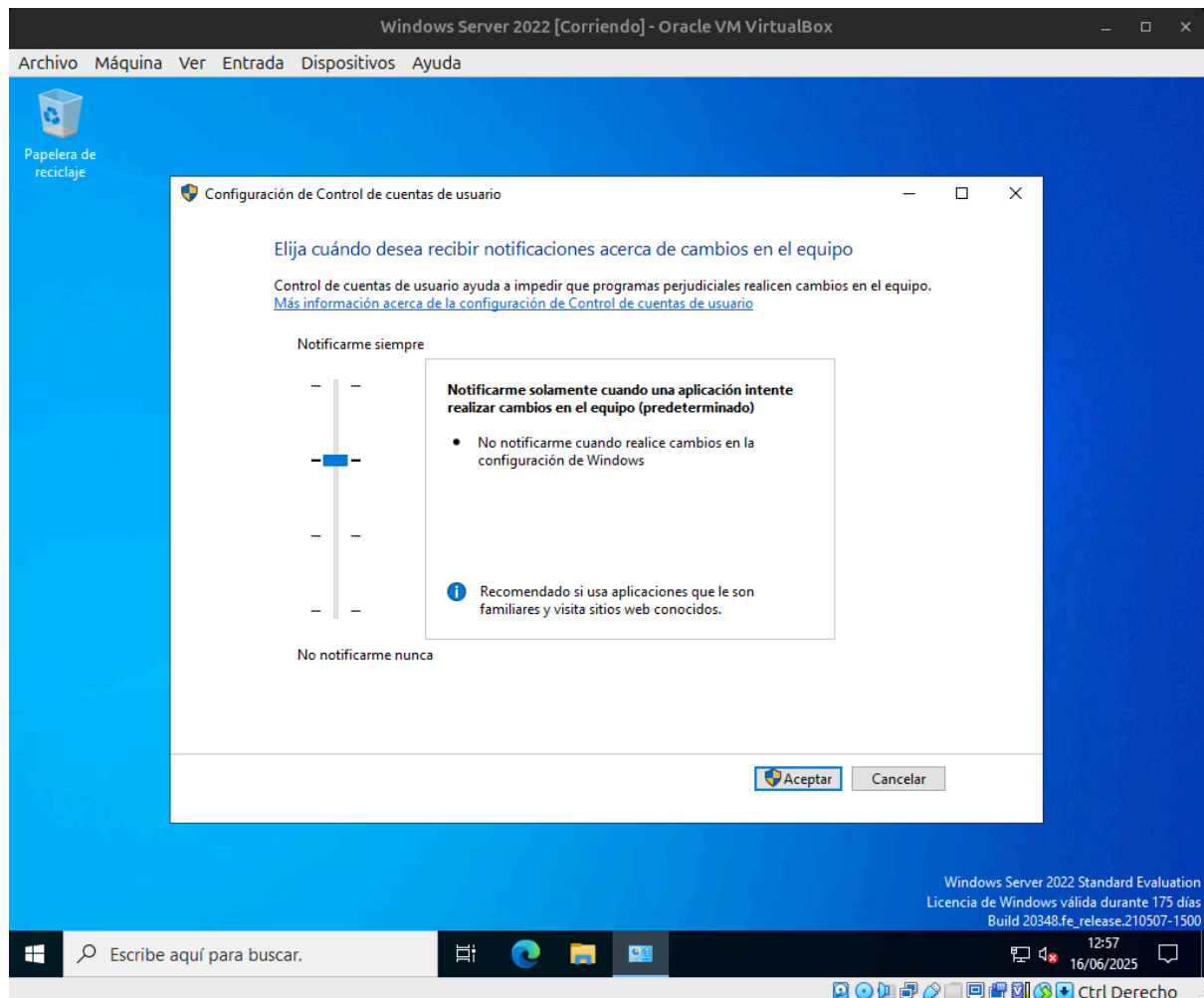


# Documentación

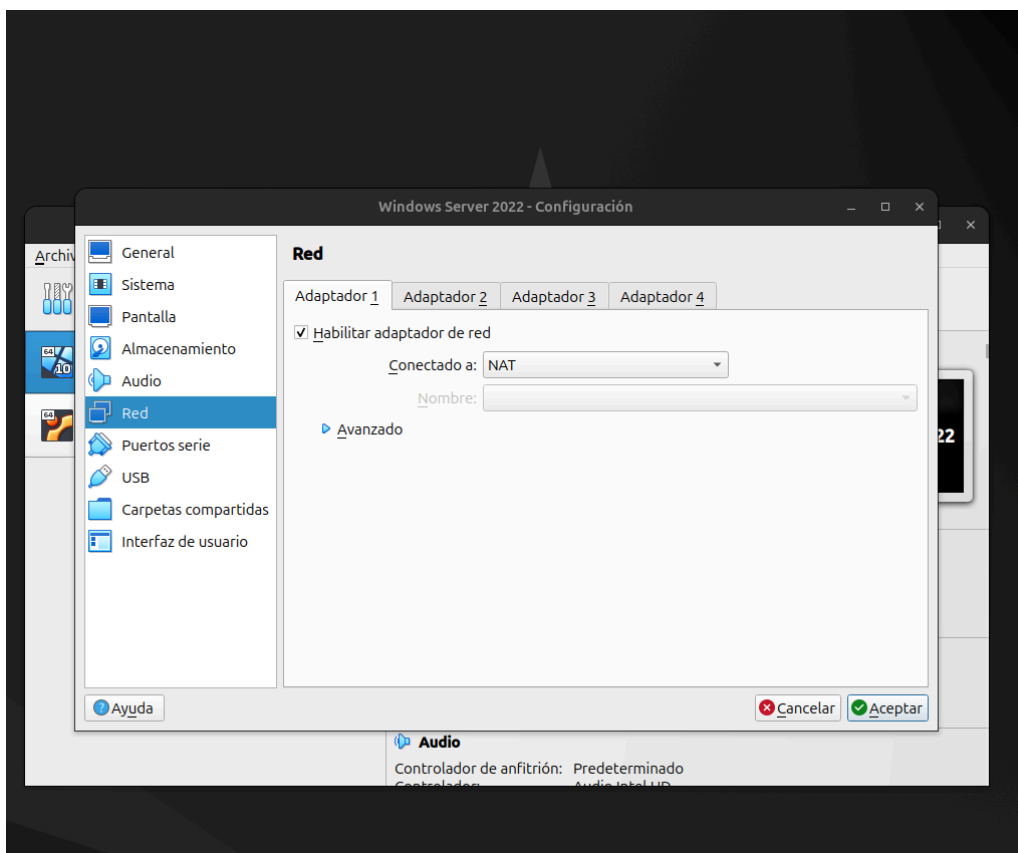
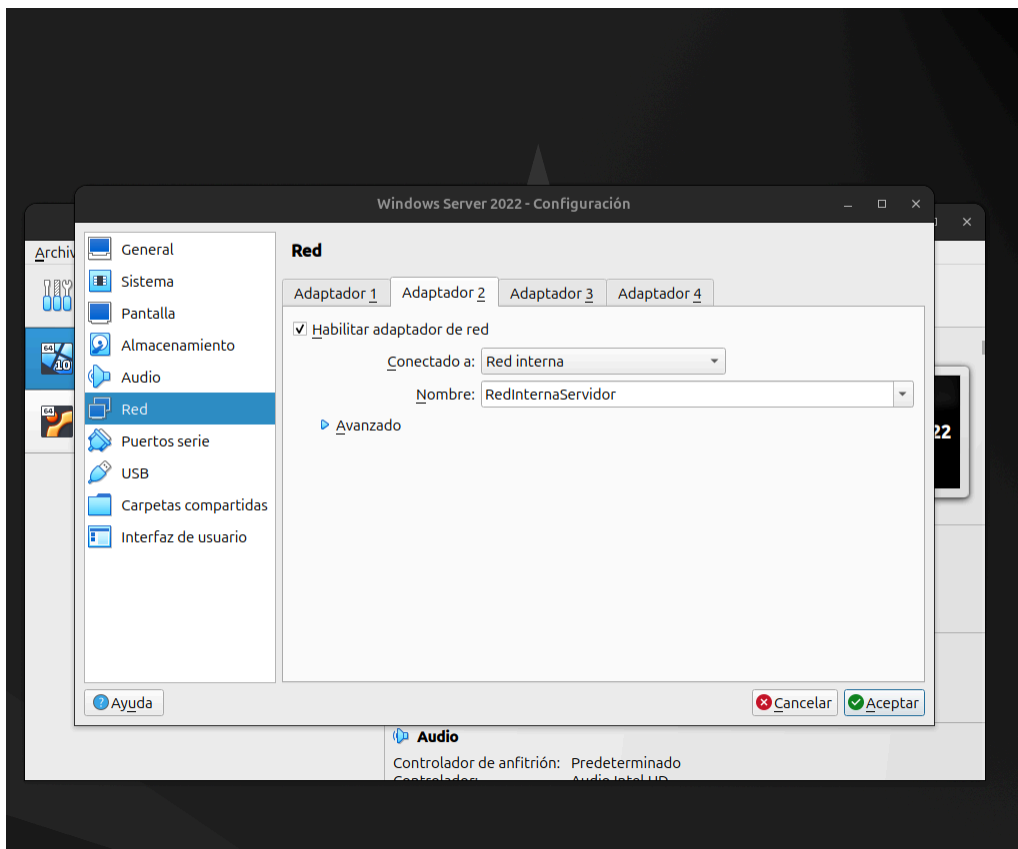
Como primer paso procedí a la creación de una cuenta de usuario con privilegios administrativos secundarios. Esta medida permite una mejor distribución de responsabilidades administrativas y una alternativa segura frente a la cuenta principal, lo que reduce la exposición ante posibles amenazas. Posteriormente, implementé una directiva de seguridad local destinada a establecer la caducidad periódica de las contraseñas de usuario, fijando un plazo de 30 días. Esta política tiene como objetivo mitigar riesgos asociados al uso prolongado de credenciales comprometidas o fácilmente vulnerables. Adicionalmente, realicé modificaciones en la configuración del Control de Cuentas de Usuario (UAC), elevando su nivel de restricción con el propósito de reforzar el control sobre la ejecución de acciones que requieran privilegios elevados, mejorando así la protección frente a cambios no autorizados en el sistema operativo.

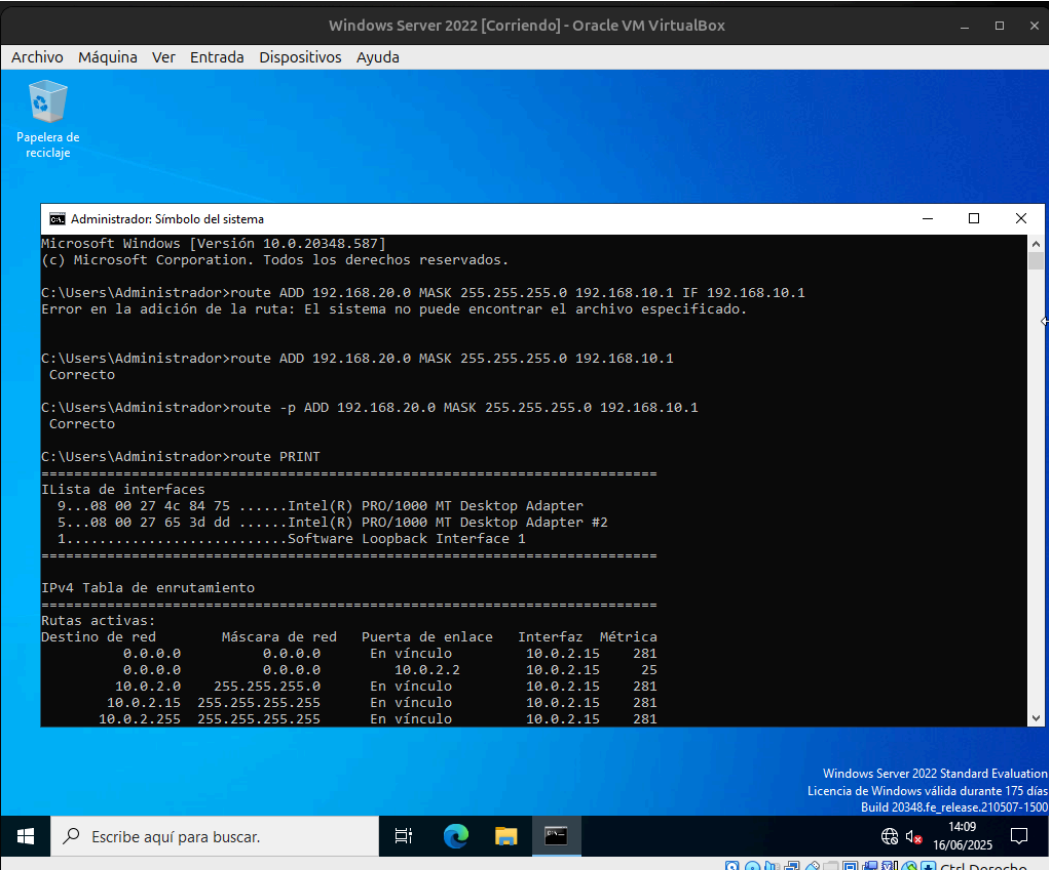
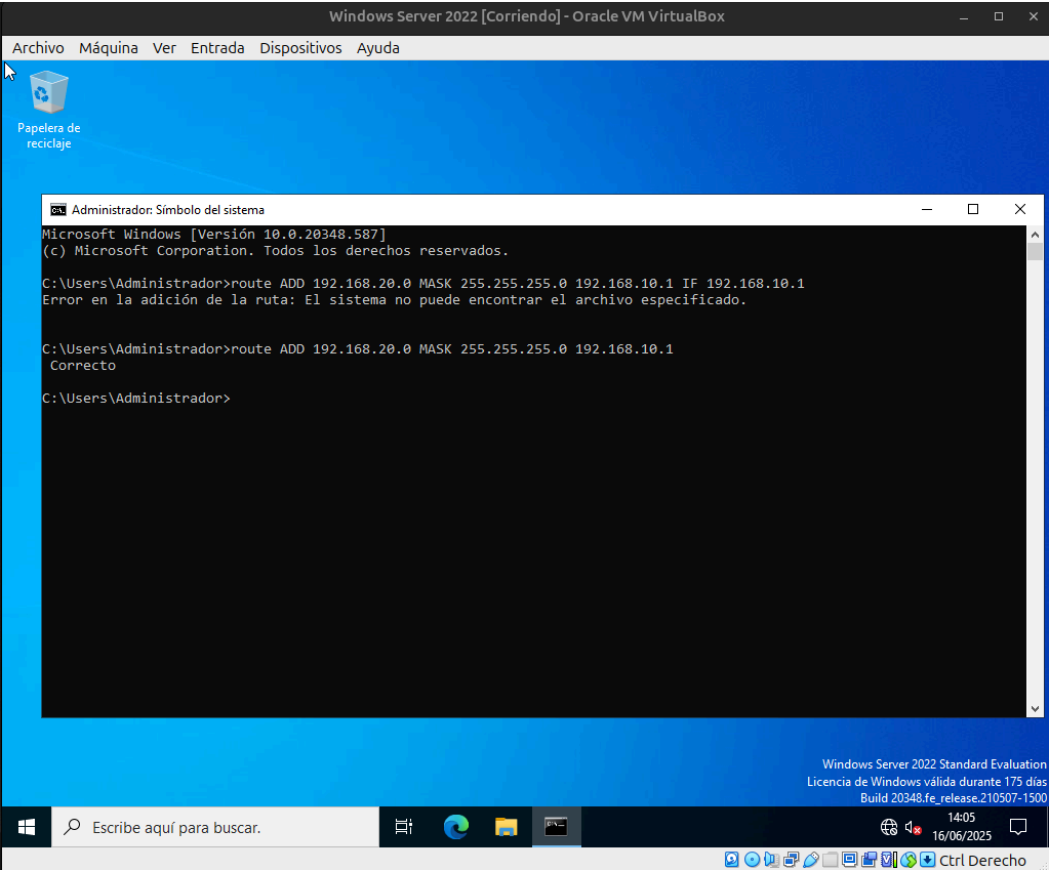


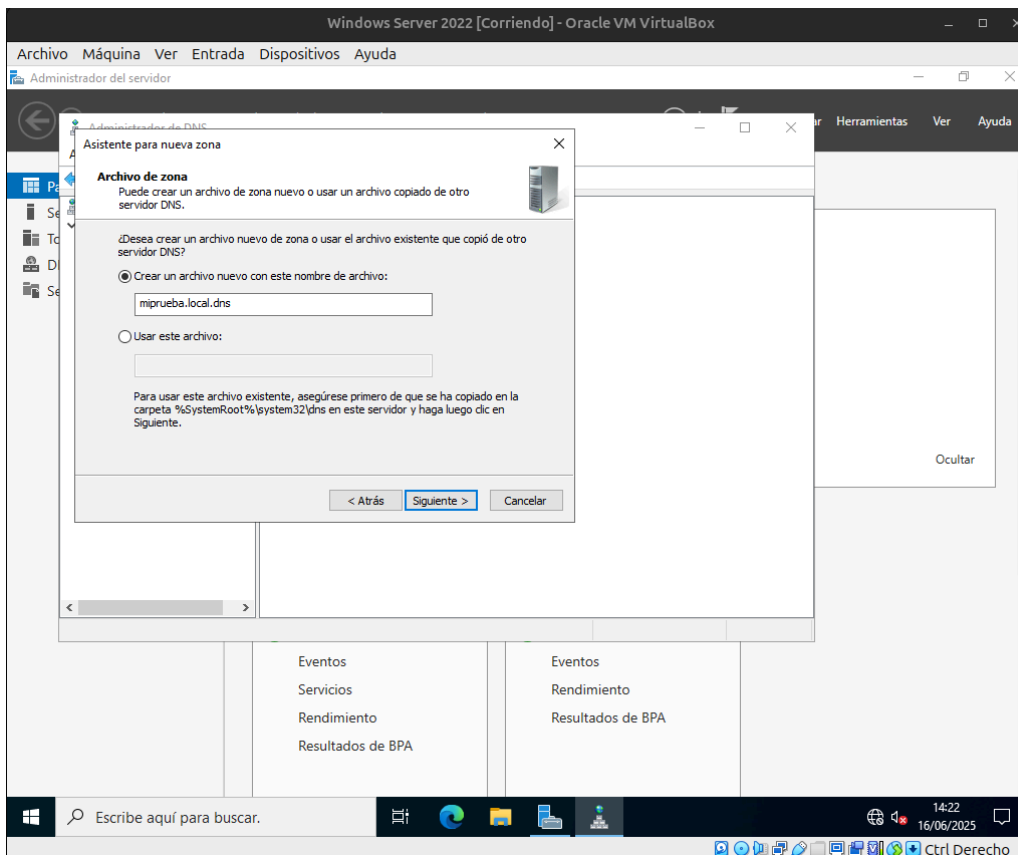
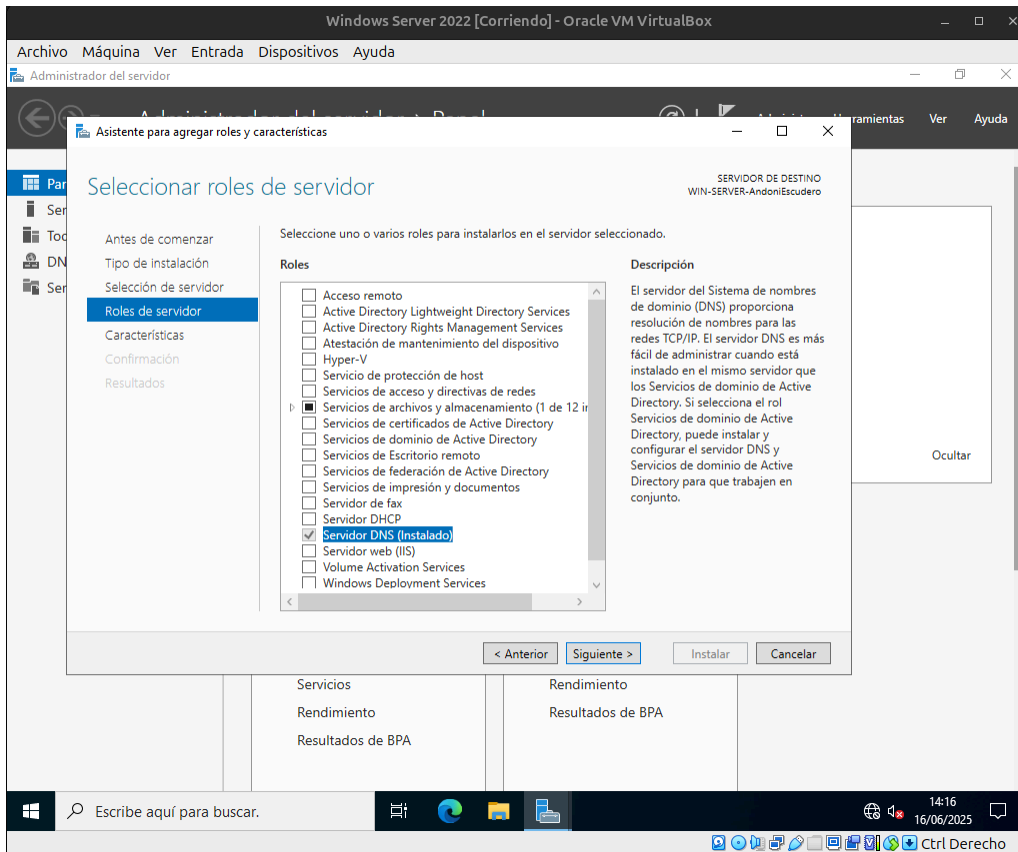


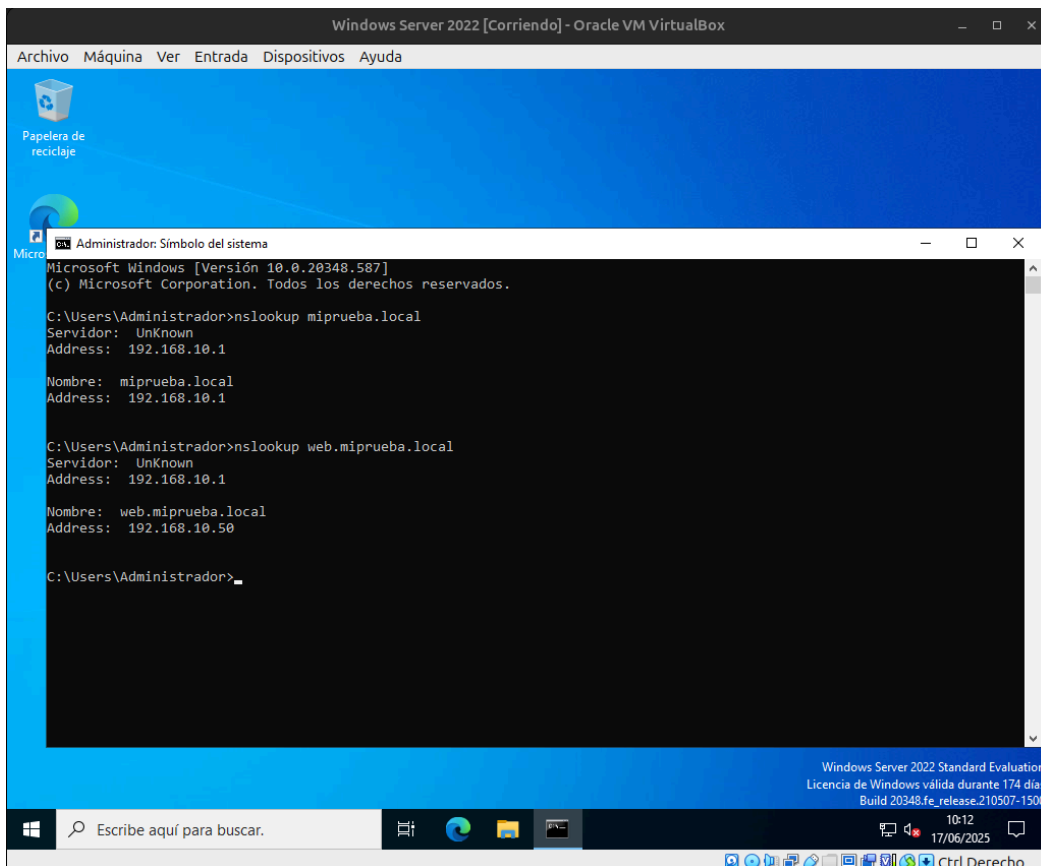
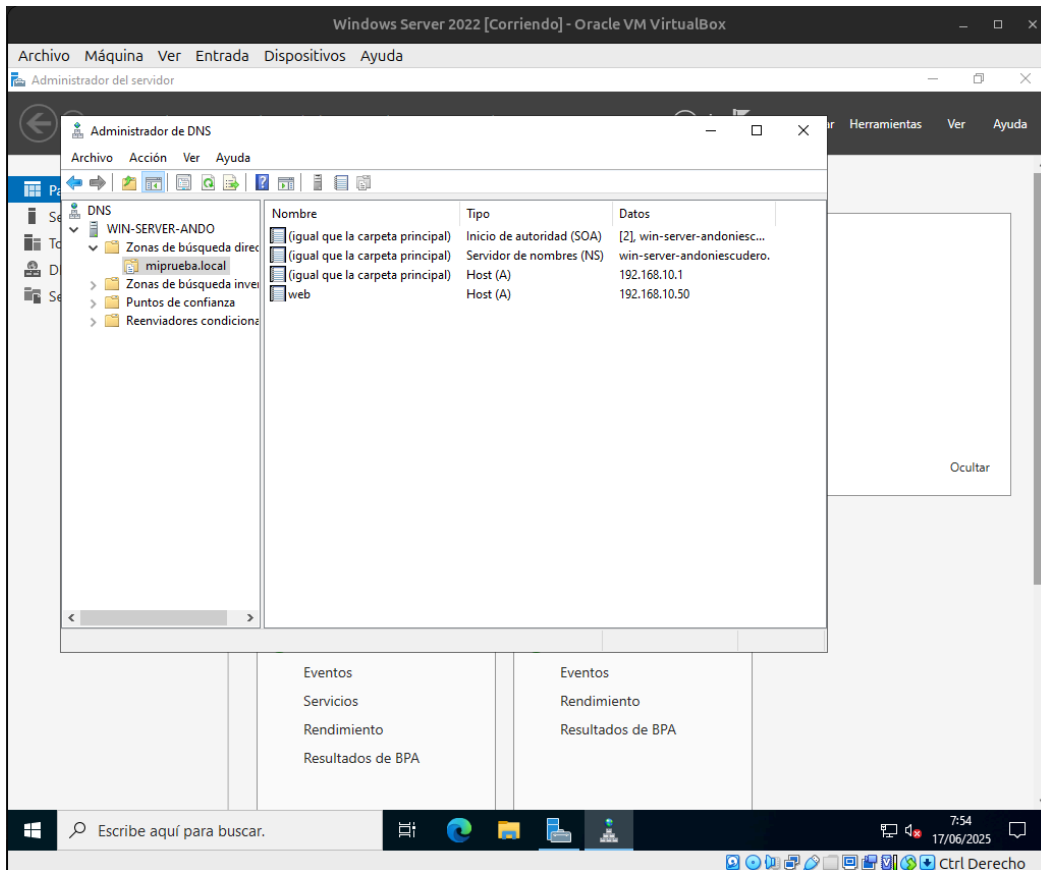


Con el objetivo de construir una infraestructura de red virtual más robusta y realista, procedí a configurar dos interfaces de red en la máquina virtual en cuestión. Paralelamente, implementé rutas estáticas en la tabla de enrutamiento del sistema operativo con el propósito de representar una topología de red más sofisticada, añadiendo complejidad al entorno simulado y posibilitando una mejor evaluación del tráfico dirigido a subredes específicas. Como complemento, instalé y puse en funcionamiento un servidor DNS local. En este servidor, configuré una zona directa que incluye al menos dos registros.

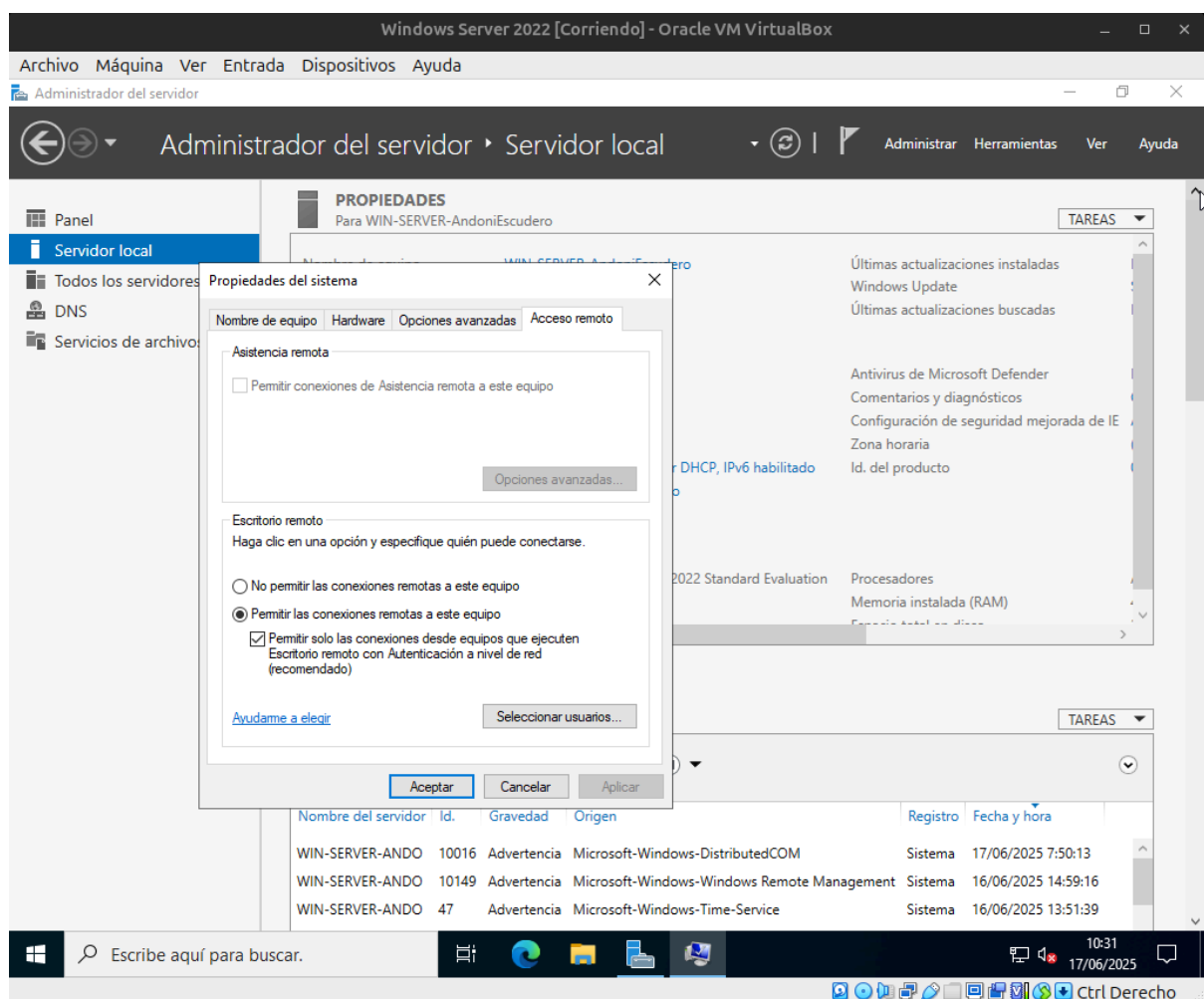






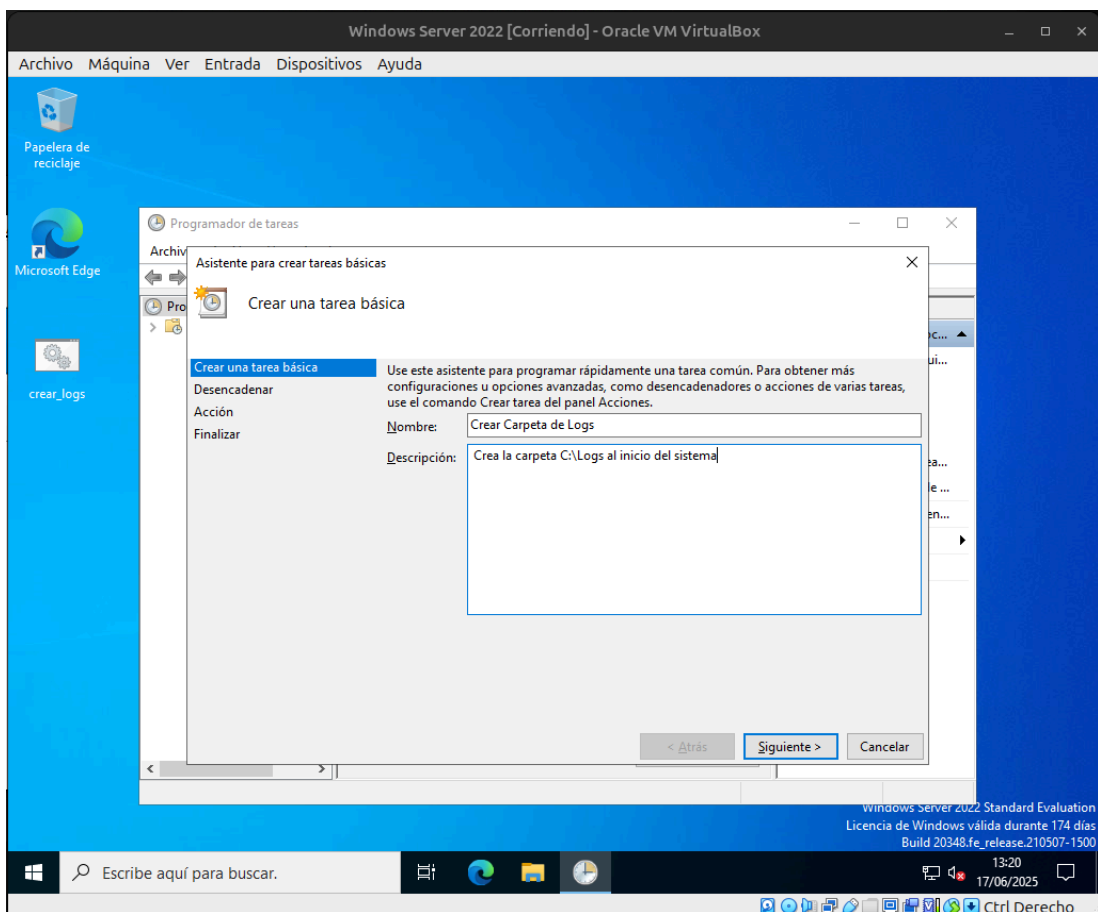
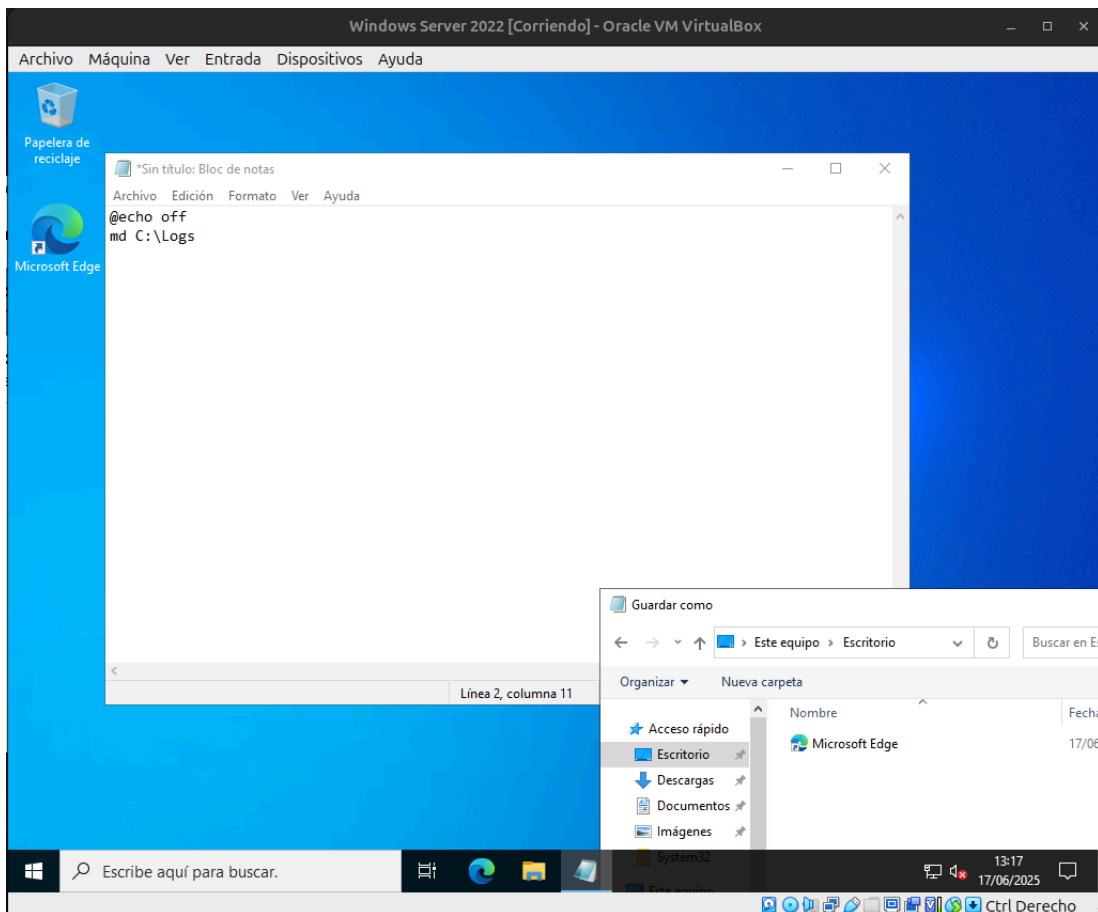


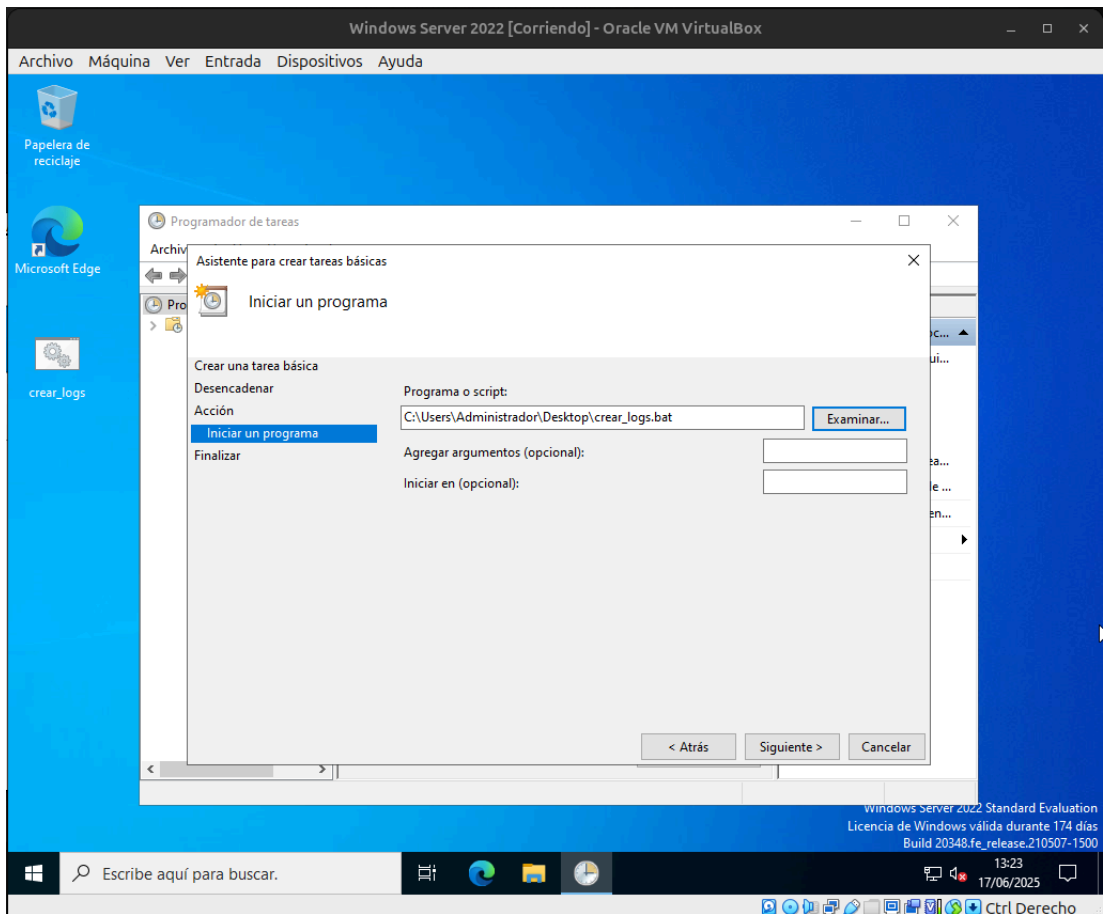
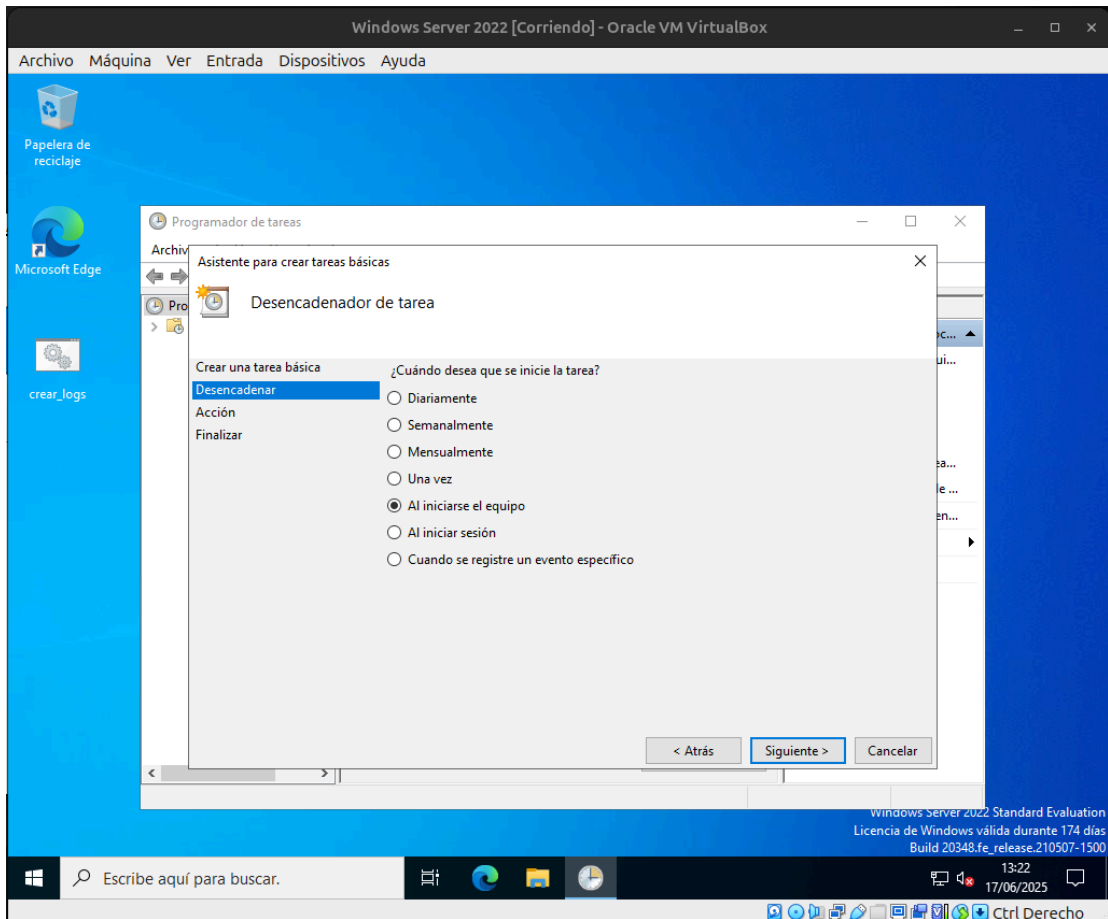
Con el objetivo de optimizar el acceso remoto al sistema manteniendo un nivel adecuado de seguridad y control, procedí a habilitar la funcionalidad de Escritorio Remoto, estableciendo una restricción explícita en el número de sesiones simultáneas permitidas, limitándolas a un máximo de dos. Asimismo, personalicé el proceso de arranque del sistema incorporando un script automatizado que, al iniciarse, verifica la existencia de una carpeta específica en la ruta C:\Logs y la crea de forma automática si no está presente. Paralelamente, realicé una configuración detallada del firewall del sistema operativo, definiendo reglas estrictas que permiten exclusivamente el tráfico de red relacionado con los protocolos RDP y DNS, bloqueando cualquier otro tipo de comunicación no autorizada. Esta política de filtrado refuerza significativamente la seguridad perimetral del entorno, reduciendo la superficie de exposición frente a posibles vectores de ataque externos.

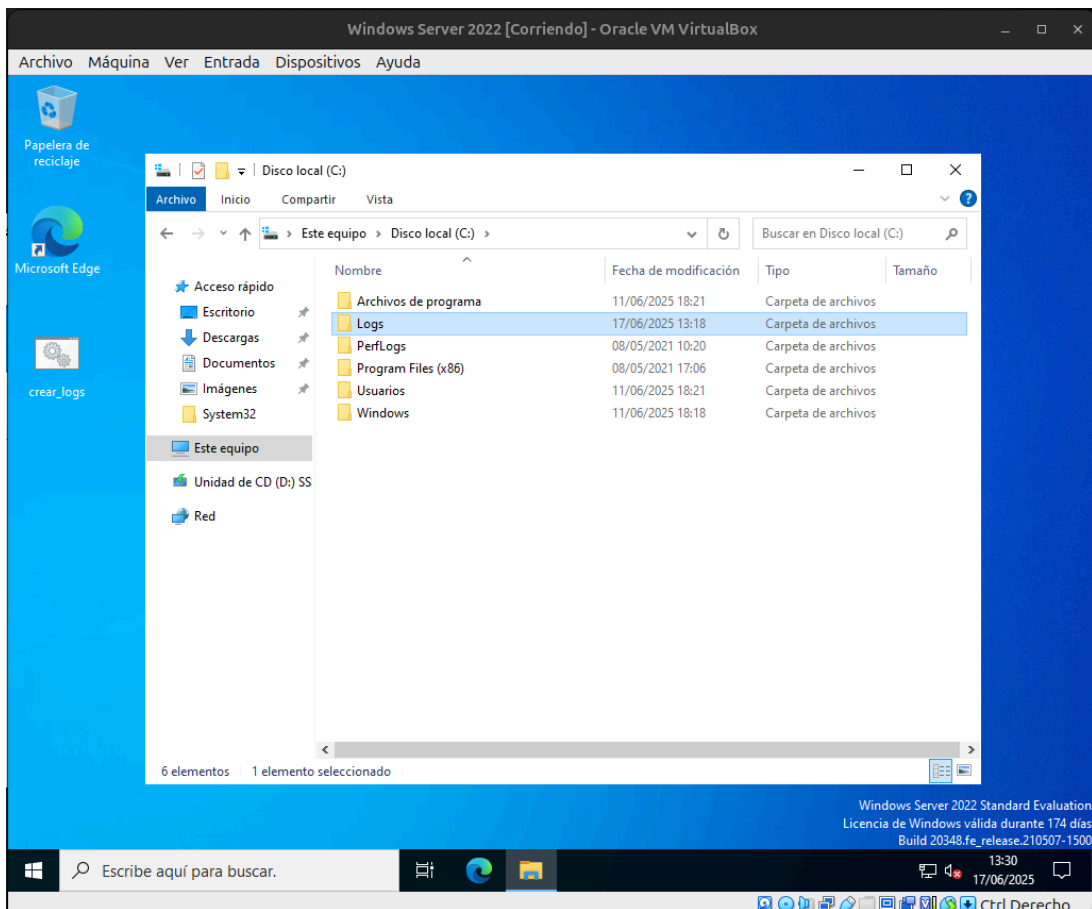
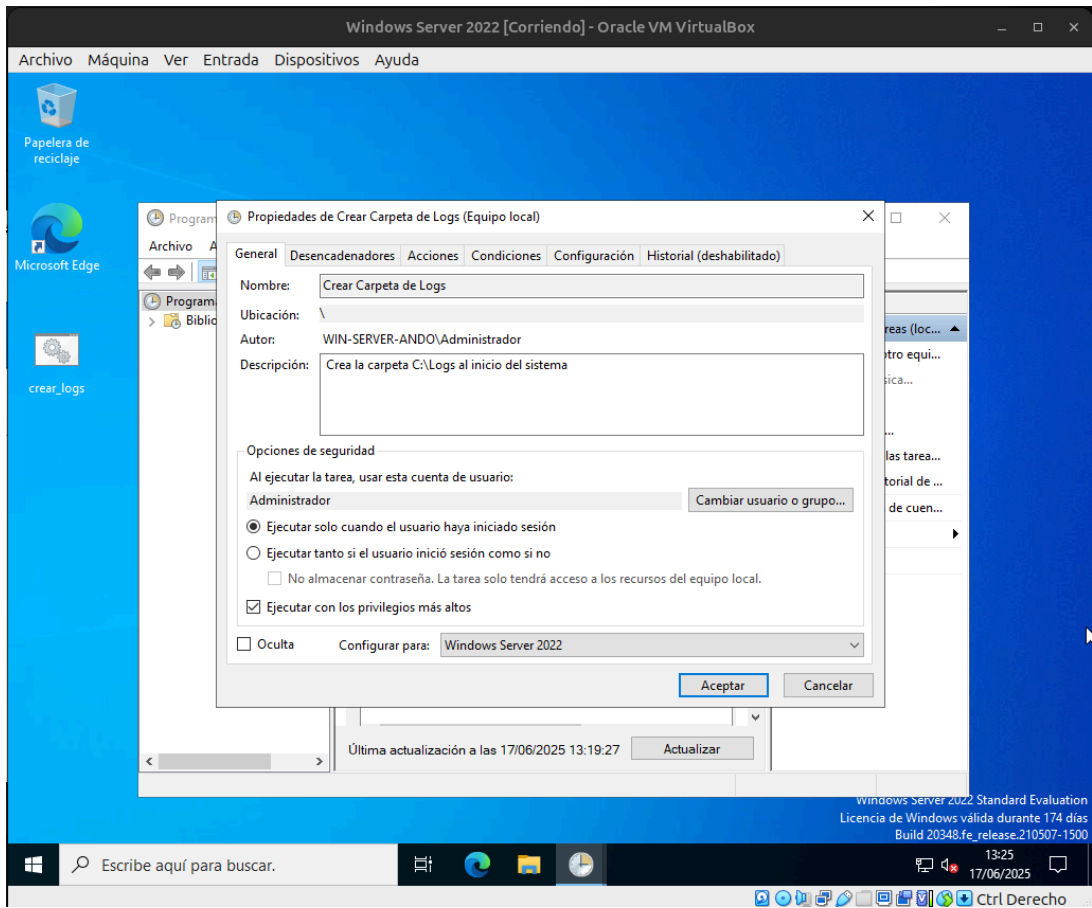


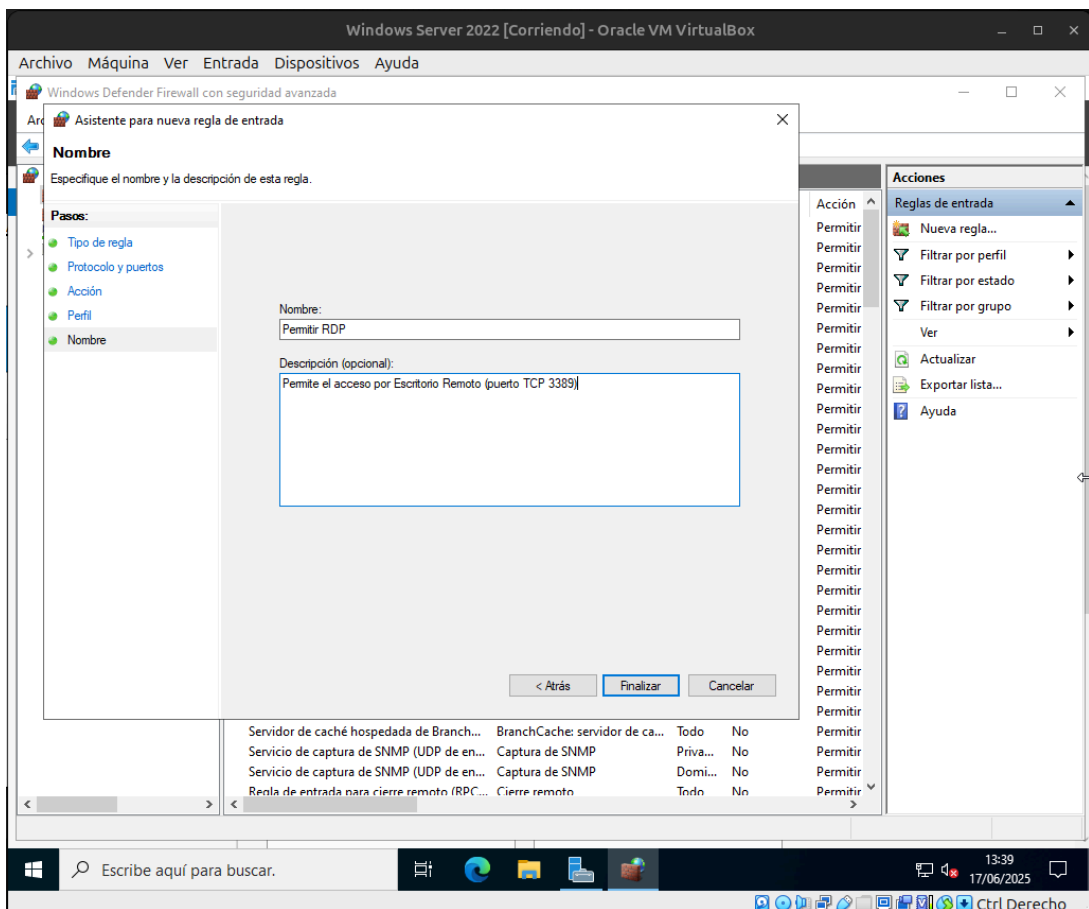
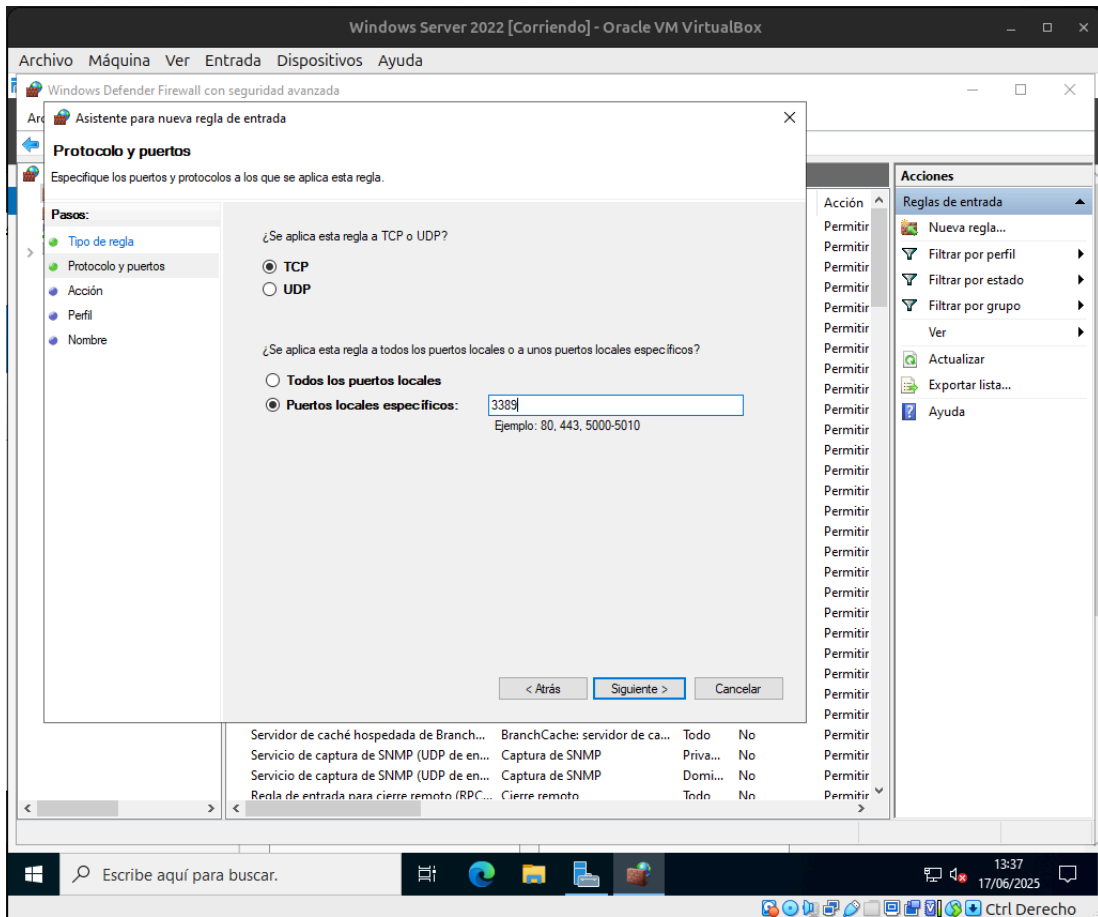


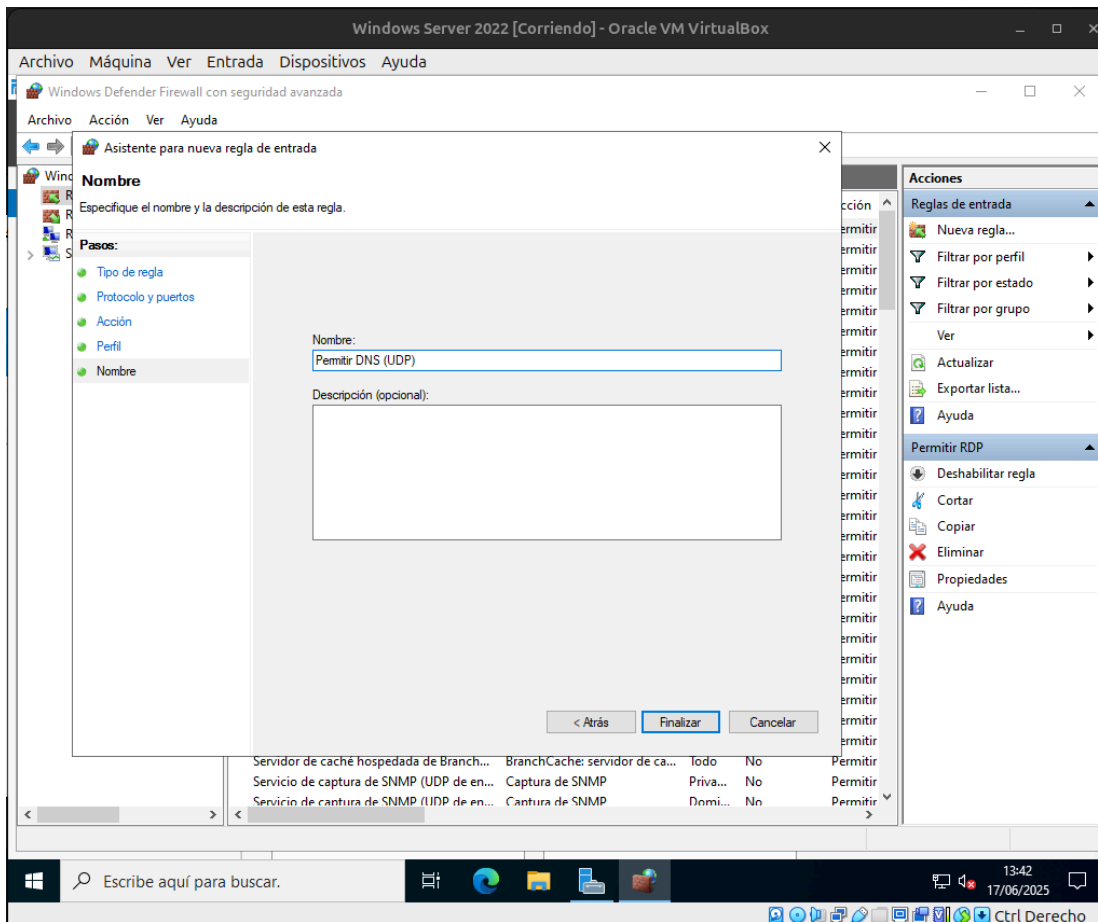
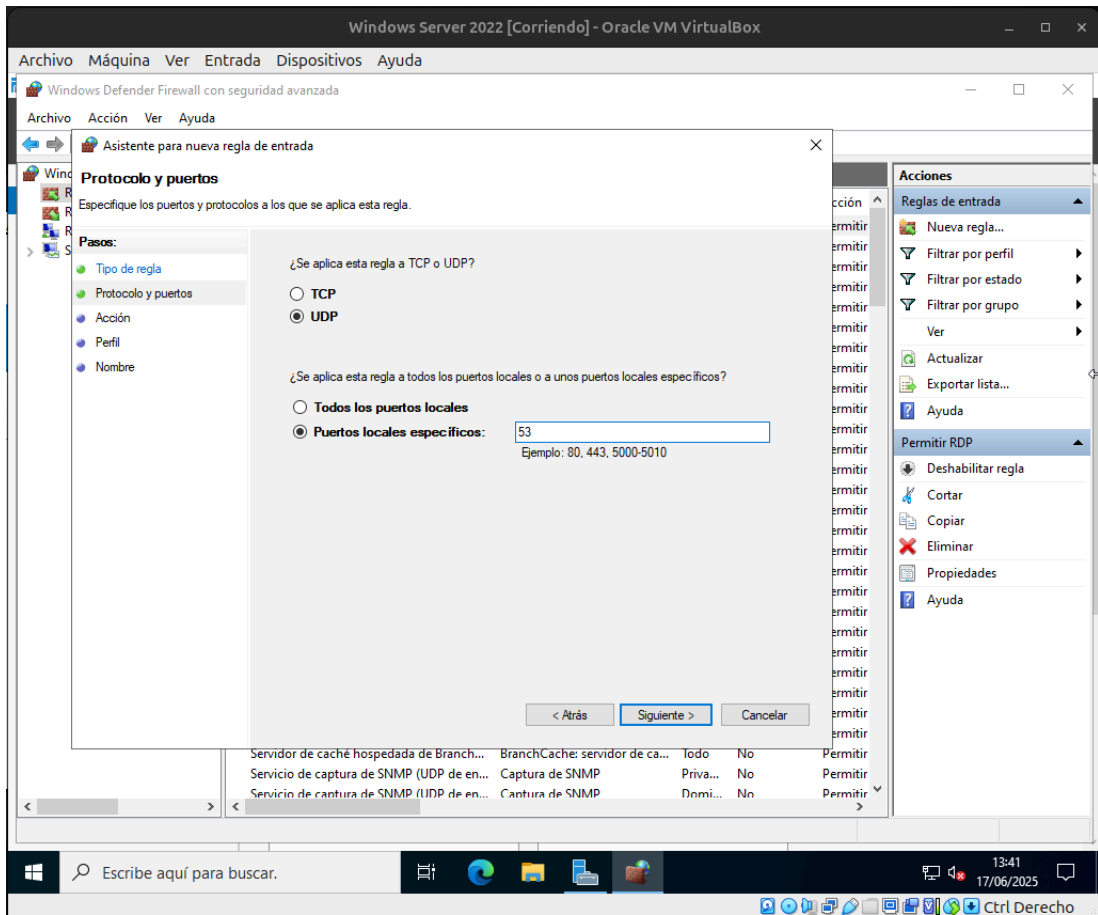


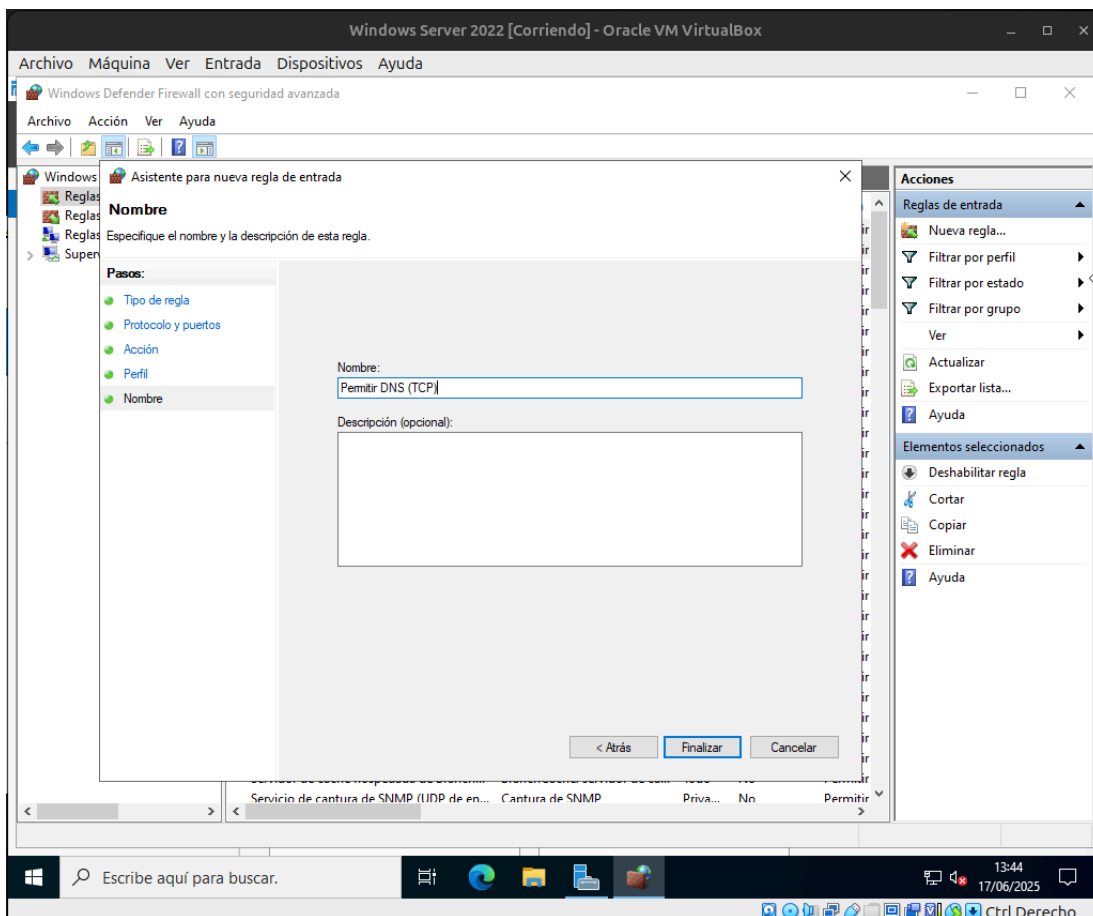
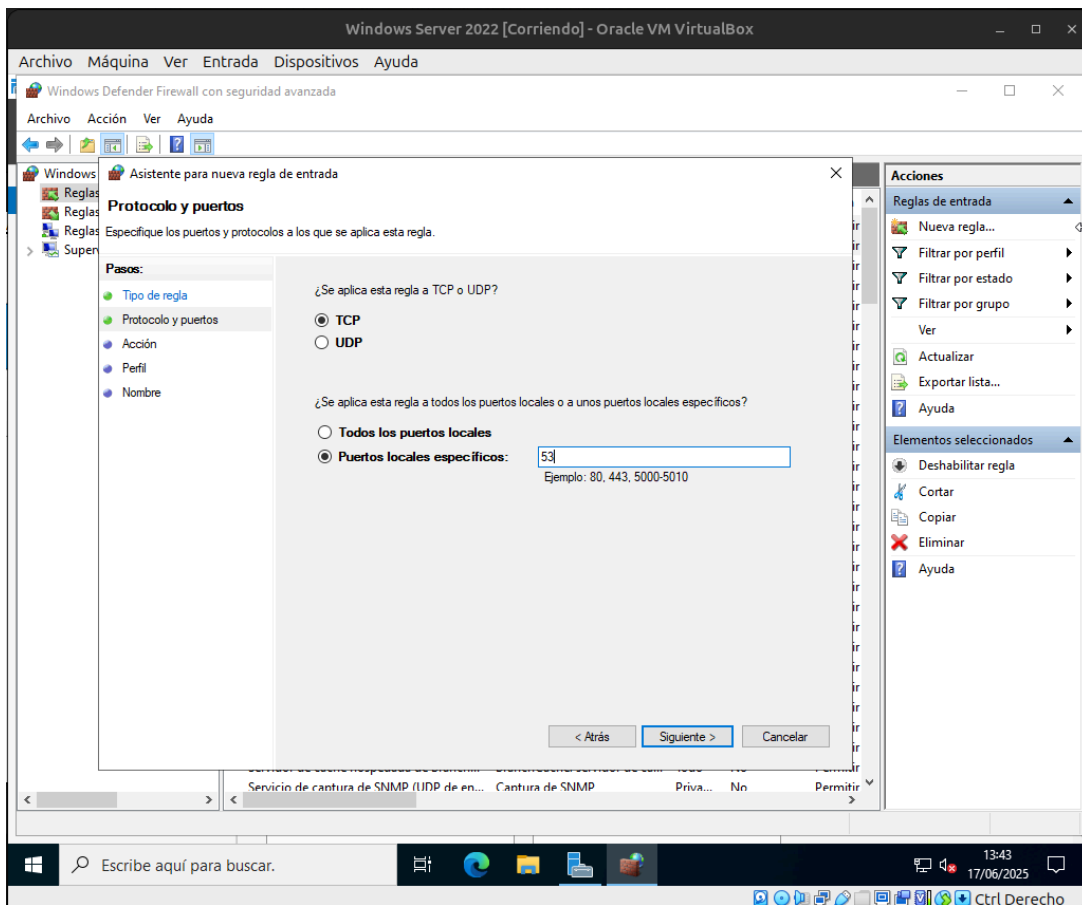


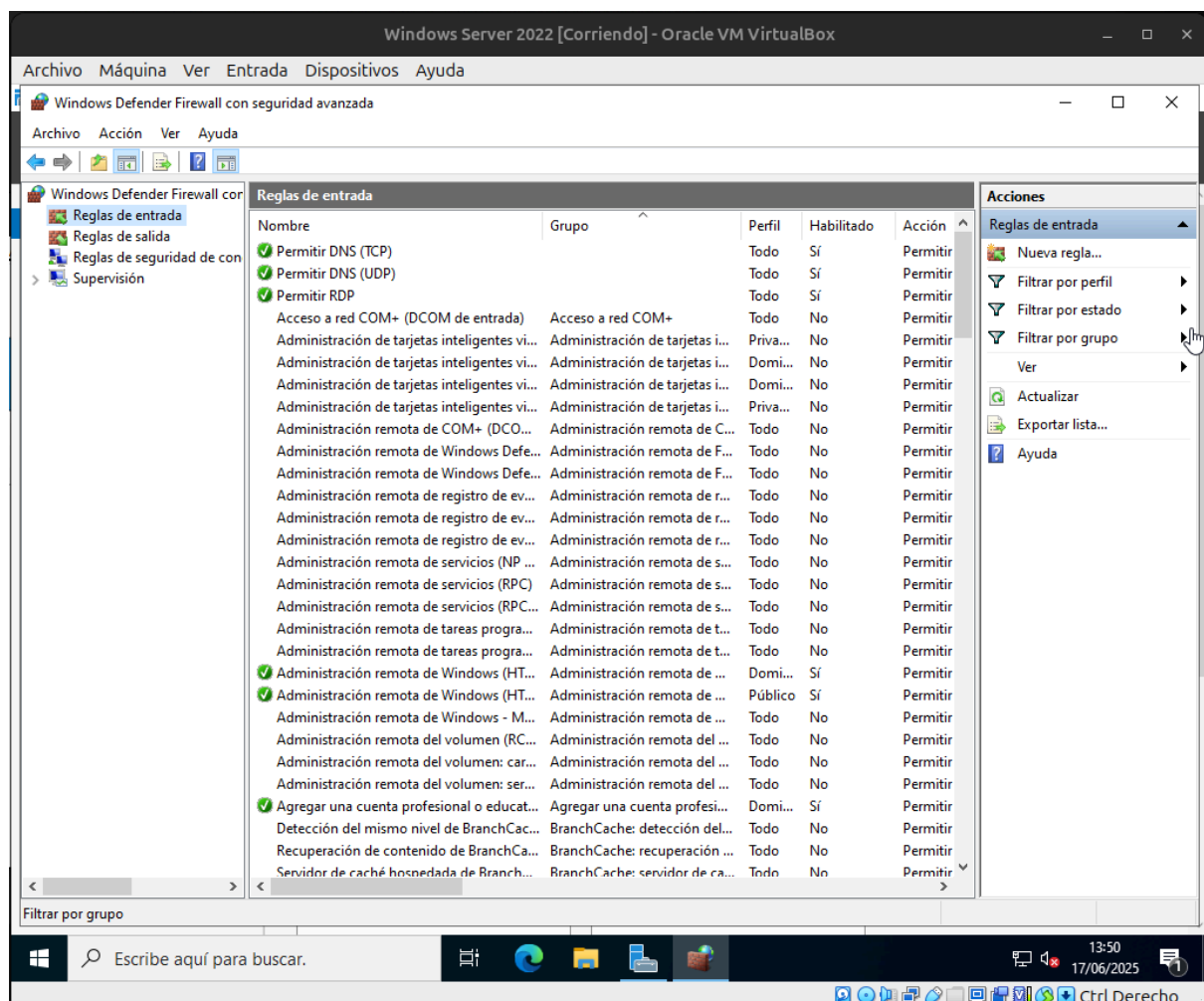






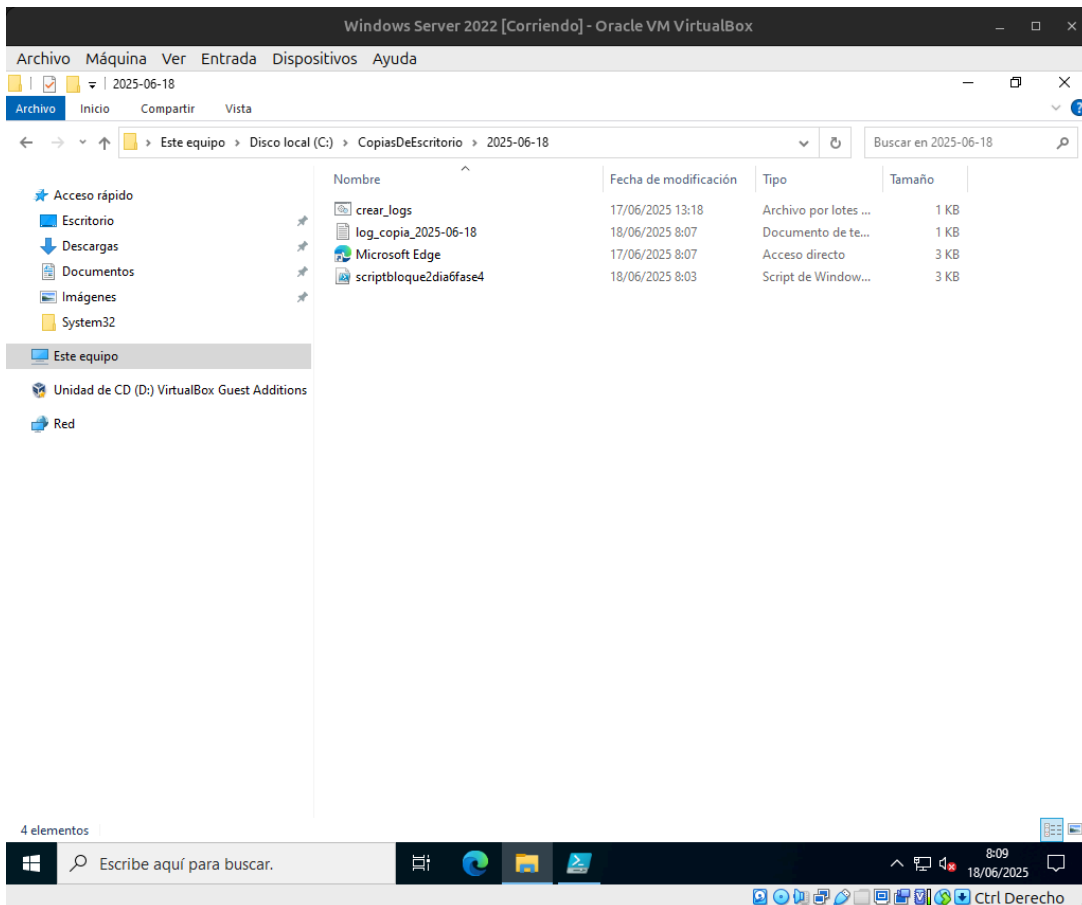
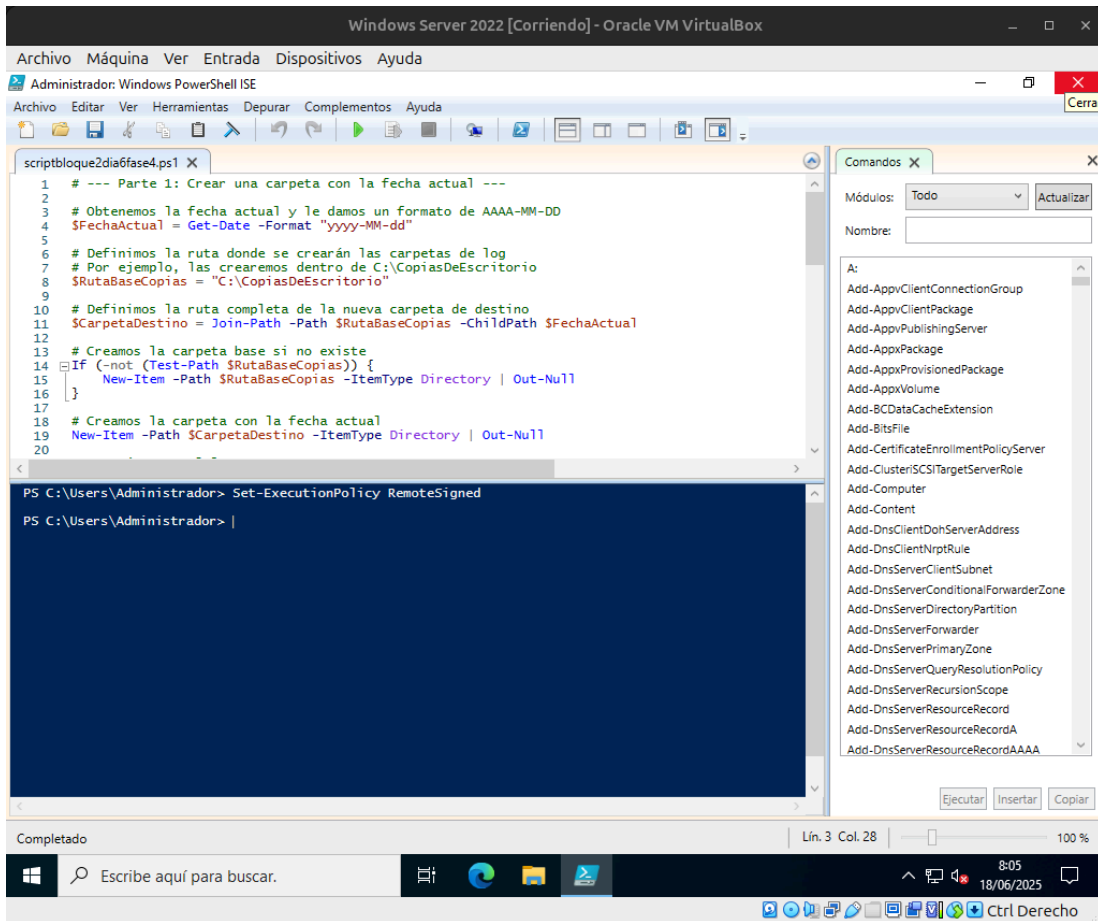


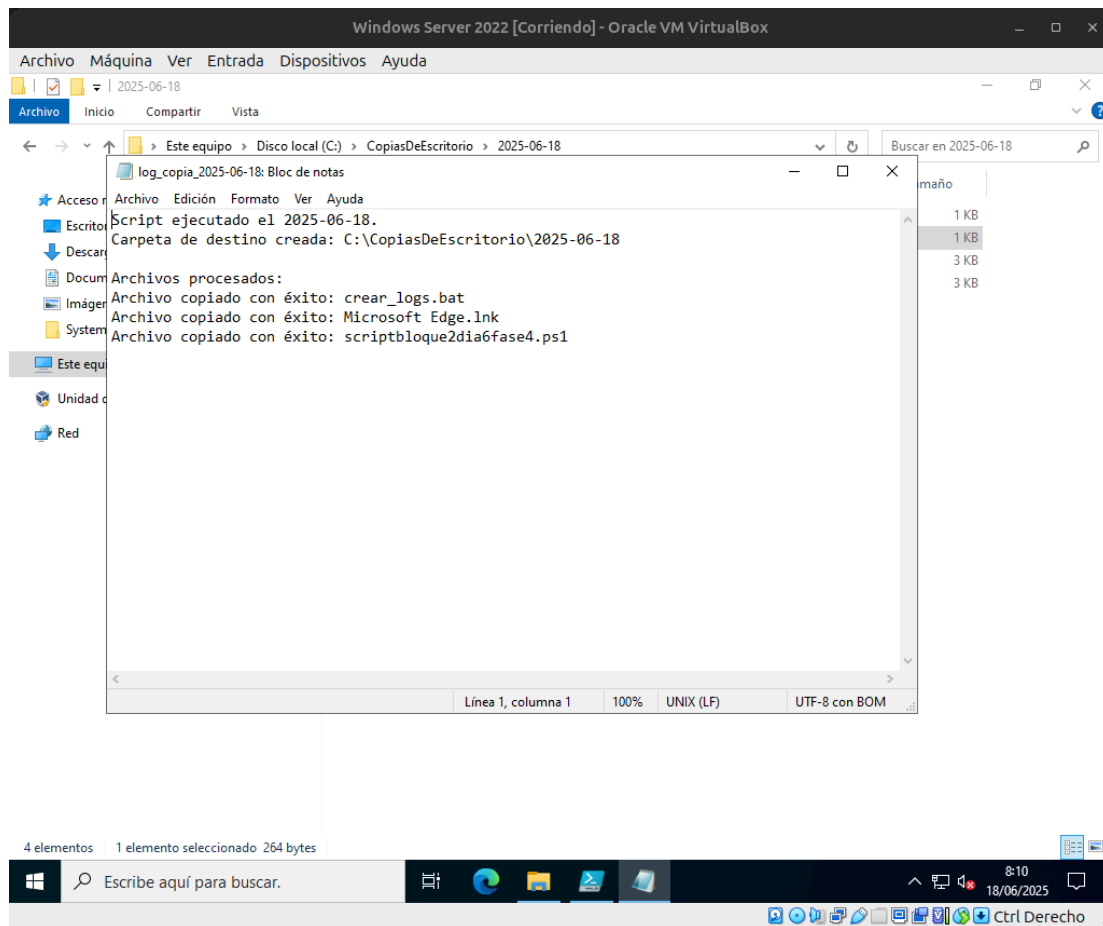




Diseñé y desarrollé un script en PowerShell orientado a la automatización de tareas rutinarias de gestión de archivos. Al ejecutarse, inicia creando de manera dinámica una carpeta cuyo nombre corresponde a la fecha actual del sistema, utilizando un formato estandarizado para facilitar su identificación cronológica. A continuación, el script procede a copiar todos los archivos ubicados en el Escritorio del usuario hacia la nueva carpeta, asegurando así una transferencia ordenada de documentos en curso o relevantes. Para garantizar la transparencia y el control del proceso, el script finaliza generando un archivo de registro con extensión .txt que documenta detalladamente el resultado de la operación de copia, incluyendo posibles errores, archivos omitidos o confirmaciones de éxito.







Código del script:

# --- Parte 1: Crear una carpeta con la fecha actual ---

# Obtenemos la fecha actual y le damos un formato de AAAA-MM-DD  
\$FechaActual = Get-Date -Format "yyyy-MM-dd"

# Definimos la ruta donde se crearán las carpetas de log  
# Por ejemplo, las crearemos dentro de C:\CopiasDeEscritorio  
\$RutaBaseCopias = "C:\CopiasDeEscritorio"

# Definimos la ruta completa de la nueva carpeta de destino  
\$CarpetaDestino = Join-Path -Path \$RutaBaseCopias -ChildPath \$FechaActual

# Creamos la carpeta base si no existe  
If (-not (Test-Path \$RutaBaseCopias)) {  
    New-Item -Path \$RutaBaseCopias -ItemType Directory | Out-Null  
}

# Creamos la carpeta con la fecha actual

```
New-Item -Path $CarpetaDestino -ItemType Directory | Out-Null
```

```
# Mensaje para el log
```

```
$LogContenido = "Script ejecutado el $($FechaActual)." + "`n"
```

```
$LogContenido += "Carpeta de destino creada: $($CarpetaDestino)" + "`n"
```

```
# --- Parte 2: Copiar archivos del Escritorio a la carpeta con la fecha ---
```

```
# Ruta de origen (el Escritorio del usuario que ejecuta el script)
```

```
$RutaOrigen = "$env:USERPROFILE\Desktop" # $env:USERPROFILE es una  
variable que apunta a C:\Users\TU_USUARIO
```

```
# Copiar archivos, excluyendo directorios (carpetas)
```

```
$ArchivosCopiados = Get-ChildItem -Path $RutaOrigen -File | ForEach-Object {
```

```
    try {
```

```
        Copy-Item -Path $_.FullName -Destination $CarpetaDestino -Force
```

```
-ErrorAction Stop
```

```
        "Archivo copiado con éxito: $($_.Name)"
```

```
    } catch {
```

```
        "ERROR al copiar archivo $($_.Name): $($_.Exception.Message)"
```

```
    }
```

```
}
```

```
$LogContenido += "`n" + "Archivos procesados:" + "`n"
```

```
$ArchivosCopiados | ForEach-Object { $LogContenido += $_ + "`n" }
```

```
# --- Parte 3: Generar un archivo de log ---
```

```
# Definimos la ruta y nombre del archivo de log
```

```
$RutaLog = Join-Path -Path $CarpetaDestino -ChildPath  
"log_copia_$(FechaActual).txt"
```

```
# Guardamos todo el contenido del log en el archivo
```

```
$LogContenido | Out-File -FilePath $RutaLog -Encoding UTF8
```

```
# Mostrar un mensaje final (opcional, si lo ejecutas manualmente)
```

```
Write-Host "Copia y log completados. Revisa: $($RutaLog)"
```