

Repository of Open Educational Resources for Laboratory Support in Engineering and Natural Science-RELAB

Project Intellectual Output 6

WEB-Laboratory: Tutorial of technical design and implementation

version 1.0, March 1st 2022

"This project has been funded with support from the European Commission. This publication reflects solely the views of the author, and the Commission cannot be held responsible for any use that may be made of the information contained therein."

INTRODUCCIÓN

El propósito de este tutorial es presentar los conceptos principales necesarios para el desarrollo de WebLabs derivados de prácticas y sistemas de laboratorio tradicionales (fuera de línea y presenciales) ya existentes.

Nuestro objetivo es proporcionar un camino para los maestros con recursos limitados y con poco o ningún soporte dedicado del departamento de TIC.

Aquí describiremos principalmente el enfoque basado en servidores de laboratorio basados en Debian de Linux, y como continuación, tutoriales adicionales muestran procesos similares para el caso en el que los servidores de laboratorio están basados en Windows.

Por lo tanto, se requiere una comprensión básica de Linux Debian OS-es y de algunos de conceptos de redes. Este enfoque debe considerarse como una alternativa a otro, descrito en una serie paralela de tutoriales, que se basa en los conceptos de EJSS y RIP-Server.

Hay algunas diferencias fundamentales en estos dos enfoques: el descrito aquí, acompañado por el documento "RELAB_Weblab_IO5_OffLine2OnLine.pdf", se basa en configuraciones de laboratorio, experimentos y prácticas tradicionales, fuera de línea, ya existentes, que deben convertirse rápidamente a versiones en línea. Mediante el uso de paquetes y servicios de software de código abierto probados y mantenidos, proponemos un método para hacer esta conversión: donde la experiencia final del usuario final proporciona conectividad de escritorio remoto, uso compartido de archivos y uso compartido del portapapeles. Por lo tanto, si los estudiantes ya están familiarizados con el uso del laboratorio "fuera de línea", podrán también hacer uso de la versión en línea con un modesto esfuerzo de aprendizaje adicional. No es un ejercicio de laboratorio preenvasado, sino más bien uno basado en la comprensión de los estudiantes de los conceptos de conectividad remota y la comprensión básica del sistema operativo Linux.

El otro enfoque, basado en el concepto de EJSS y RIP-Server, es un enfoque más simplificado que requiere un esfuerzo adicional por parte del profesor para preparar tanto la configuración del laboratorio como la aplicación de laboratorio front-end. Además, deben usarse bibliotecas personalizadas en el lado de la plataforma Arduino o RIP para habilitar los métodos RPC: recopilar variables de entrada de control y contabilizar variables de salida (por ejemplo, lectura de sensores o parámetros internos). Este es un enfoque más predefinido que permite a los estudiantes explorar conceptos bien definidos sin ninguna comprensión profunda de la conectividad remota y los procesos en el servidor que aloja el laboratorio. En ese sentido, es mejor experiencia para ejercicios puramente enfocados en, por ejemplo, teoría de control, procesamiento de señales o similares. Los estudiantes podrían acceder a la aplicación del laboratorio solo a través del navegador. Sin embargo, se requerirían pasos adicionales para acceder a los laboratorios sin dirección IP pública/estática (lo que no se describe específicamente en el conjunto actual de tutoriales).

Configuraciones de laboratorio accesibles de forma remota - WebLab

WebLab es un concepto introducido recientemente que describe (típicamente) sistemas físicos de laboratorio a las que se puede acceder a través de la Internet pública. Existe una línea borrosa de distinción entre los laboratorios accesibles remotamente sobre la infraestructura de intranet del campus y los laboratorios web.

La motivación para la accesibilidad remota de los laboratorios sobre el campus suele ser diferente de la que hay detrás de los WebLabs: puede venir dada por restricciones de oficina, razones de seguridad y/o razones de comodidad / productividad. Dado que los servicios VPN modernos ofrecen experiencias casi idénticas, dado que el ancho de banda de la red y las latencias son suficientes, el acceso público y la intranet pueden considerarse lo mismo.

Por lo tanto, nos centraremos en enfoques alternativos, que requieren menos o ningún soporte del departamento de TIC y más adecuados para una audiencia más amplia de estudiantes. Aun así, las preocupaciones de seguridad deben abordarse tanto en términos de intrusiones externas, acceso inadvertido de los estudiantes a los recursos del laboratorio no destinados a las prácticas, así como la interferencia mutua de las actividades entre los estudiantes.

Dado que existe una superposición significativa en la implementación y descripción del diseño tecnológico de WebLabs, este documento debe usarse junto con "RELAB_WebLab_IO5_OffLine2OnLine.pdf".

Los principales servicios adicionales que se proporcionarán, en comparación con el enfoque de servidor de laboratorio "fuera de línea", son:

1. Conectividad de escritorio remoto
2. Intercambio de archivos / Intercambio de portapapeles
3. Transmisión de vídeo (opcional), unidireccional (laboratorio → escritorio del alumno)

Arquitectura WebLab de RELAB

La arquitectura WebLab incluye 3 elementos.

1. **El programa de instalación del laboratorio** debe incluir el equipo que se usará por dos motivos:
 1. Para el control de equipos conectados localmente
 2. Como nodo de comunicación permitiendo así el acceso remoto.

Además de la computadora, el equipo donde se realizan los experimentos reales es también parte del laboratorio. Este equipo debe incluir mecanismos de control remoto, hoy en día típicamente basados en puertos serie (RS232), comunicación USB, acceso LAB 100/1000Mbps, WiFi o PCIe. Obviamente, un equipo que solo tiene controles / pantallas manuales no es adecuado para este propósito. Ejemplos notables son varios tipos de placas Arduino, ESP32, RaspberryPi 3 o 4, y muchos otros instrumentos de laboratorio con control remoto incorporado.

Si ciertos instrumentos o fenómenos físicos necesitan ser observados en tiempo real, una cámara web (conectada por USB2/3), debe estar disponible desde el servidor del laboratorio.

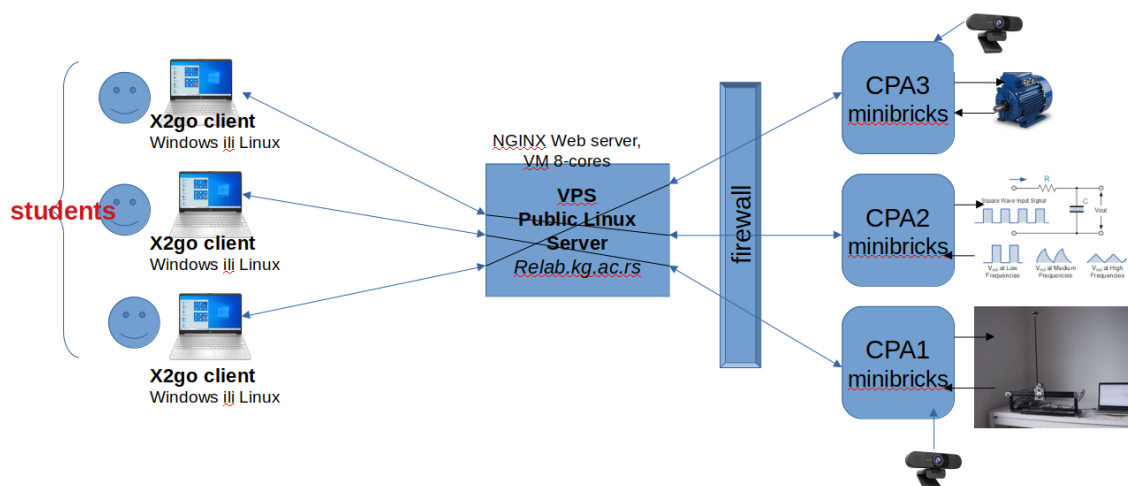
1. **WebLab Public (también conocido como Bastión) Server** es típicamente VPS (Virtual Private Server). No es necesario que se dedique a este único propósito, ya que se recomienda encarecidamente utilizar la distribución Linux de código abierto como sistema operativo VPS. En el resto del documento, asumimos que esta es una de las distribuciones basadas en Debian, más específicamente, Ubuntu. Este es un servicio público que permite la exposición de activos seleccionados detrás del firewall. De esta

manera, las posibilidades conexión se reducen significativamente a un solo VPS, de acuerdo a los métodos estándar de protección que deben seguirse estrictamente. Además del estricto servicio "Bastion" (también conocido como "Proxy Jump"), con respecto a WebLab, este VPS puede alojar muchas otras aplicaciones web, como, por ejemplo, una para permitir reservar fechas y horas para acceder al laboratorio y monitorizar los accesos al mismo.

2. **La estación de trabajo del cliente/estudiante** debe incluir muy pocas modificaciones de software. Puede ser estándar, basado en Windows o Linux. Este PC debe tener requisitos de rendimiento moderados. Se utilizará principalmente como terminal de escritorio remoto, con disposiciones para el intercambio de archivos, principalmente para recopilar datos o pruebas de que se ha realizado el ejercicio de laboratorio. O bien, puede ser que se use para transferir fragmentos de código o archivos de configuración que podrían haberse preparado sin conexión.

Idealmente, se desea que no se necesiten paquetes de software adicionales, lo que se puede lograr utilizando navegadores locales modernos. No obstante, la instalación simple de paquetes de software adicionales que lleve menos de 5 minutos también es aceptable.

Arquitectura de ejemplo para escritorio remoto y uso compartido de archivos:



Alternativas importantes: Apache Guacamole

Este software ofrece una puerta de enlace de escritorio remoto de código abierto (<https://github.com/apache/guacamole-server>) (multiplataforma) que admite SSH, VNC y RDP. Tiene una implementación de servidor que puede ejecutarse en cualquier distribución moderna de Linux, y un cliente que puede ejecutarse en cualquier navegador moderno. También tiene una aplicación basada en web que puede proporcionar acceso al servidor del laboratorio, conteo de usuarios y monitorización de actividades. Este servicio se puede instalar en el servidor 'Bastion', como uno adicional, en paralelo con la solución propuesta y actualmente verificada.

Como cualquier otra solución, esta puede sufrir vulnerabilidades, por lo que se requieren actualizaciones frecuentes para mantener las versiones actualizadas.

Transmisión de video:

Existen varios métodos para lograr este servicio de generación de transmisión de video en el servidor del laboratorio y ponerlo a disposición del lado del estudiante en tiempo real.

Un buen ejemplo a seguir se proporciona en: <https://www.nginx.com/blog/video-streaming-for-remote-learning-with-nginx/>

Se basa en FFMPEG o VLC, utilizado en el lado del servidor de laboratorio para tomar la transmisión de video de la cámara web (recibida por el servidor de laboratorio a través de la conexión USB) y convertirla en transmisión RTMP (o similar, por ejemplo, .SRT) al mismo servidor VPS que aloja "Bastión". El servicio web adicional obtiene la secuencia y la convierte en una secuencia MPEG-DASH de archivos que puede ser observada por varios usuarios. Esto da la oportunidad de tener varios estudiantes conectados simultáneamente y viendo el mismo experimento. Un inconveniente de la transmisión MPEG-DASH / HLS es la latencia inherente de 2-3 segundos, que puede evitar la interactividad de alta velocidad.

Para una latencia más baja, si el servidor de laboratorio tiene una dirección pública (X.Y.Z.W) y la transmisión uno a uno es suficiente, se puede lograr rápidamente con FFMPEG en el lado del servidor de laboratorio y el reproductor VLC (debe instalarse en el PC del estudiante, <https://github.com/videolan/vlc>):

```
ffmpeg -f video4linux2 -framerate 25 -video_size 1280x720 -i /dev/video0 \
  -f alsa -ac 2 -i sysdefault:CARD=WEBCAM -c:v libx264 -b:v 1600k -preset ultrafast \
  -x264opts keyint=25 -g 25 -pix_fmt yuv420p -c:a aac -b:a 128k \
  -vf "drawtext=fontfile=/usr/share/fonts/truetype/dejavu/DejaVuSansMono.ttf: \
text='CLOUD DETECTION CAMERA 01 UTC-3 {%localtime\:%Y-%m-%dT%T}': fontcolor=white@0.6: fontsize=12: x=10: y=10: box=1: \
boxcolor=black: boxborderw=6" \
  -f rtp_mpegts "rtp://X.Y.Z.W:5111?ttl=2"
```

En el lado del PC del estudiante, tendremos VLC player, con conexión abierta a: "rtp://X.Y.Z.W:5111".

Otro enfoque para permitir la transmisión de video de menor latencia está basado en el protocolo SRT. Esta es una adición emergente más reciente (por ejemplo, en comparación con RTMP), que garantiza menores latencias de transporte. Está basado en UDP, y se pueden encontrar más detalles en <https://github.com/Haivision/srt>.

Tanto FFMPEG como VLC tienen soporte para protocolos SRT en versiones más recientes, o se pueden habilitar como se describe en <https://srtlab.github.io/srt-cookbook/apps/ffmpeg/> y <https://srtlab.github.io/srt-cookbook/apps/vlc-media-player/>

Desde la configuración del servidor de laboratorio, la transmisión se puede enviar con (si ffmpeg se compila con --enable-libsrt):

```
ffmpeg -f video4linux2 -framerate 25 -video_size 1280x720 -i /dev/video0 \
  -f alsa -ac 2 -i sysdefault:CARD=WEBCAM -c:v libx264 -b:v 1600k -preset ultrafast \
  -x264opts keyint=25 -g 25 -pix_fmt yuv420p -c:a aac -b:a 128k \
  -vf "drawtext=fontfile=/usr/share/fonts/truetype/dejavu/DejaVuSansMono.ttf: \
text='CLOUD DETECTION CAMERA 01 UTC-3 {%localtime\:%Y-%m-%dT%T}': fontcolor=white@0.6: fontsize=12: x=10: y=10: box=1: \
boxcolor=black: boxborderw=6" \
  -f mpegts "srt://X.Y.Z.W:5112?mode=caller&passphrase=mypasswd&pbkeylen=32&oheadbw=100"
```

En el lado del PC del estudiante tendremos VLC player, con conexión abierta a: "srt://X.Y.Z.W:5112". Las opciones adicionales como contraseña, latencia, longitud de clave deben especificarse en la GUI de la aplicación VLC, como se describe en <https://srtlab.github.io/srt-cookbook/apps/vlc-media-player/>

Detalles de configuración del servidor de laboratorio

La configuración del servidor de laboratorio puede basarse en un PC ligero o de rendimiento medio. Se recomienda que al menos dos núcleos de CPU x86 estén disponibles con al menos 4 GB de DDR y espacio en disco moderado. Los requisitos adicionales de rendimiento pueden provenir de cargas de trabajo que se ejecutan localmente, pero para el acceso remoto anterior, los requisitos moderados son suficientes. Vale la pena mencionar que una tarjeta gráfica con GPU externa no sirve para este propósito. Sin embargo, si el servidor de laboratorio no está dedicado permanentemente a este rol, es posible que se aplique un conjunto diferente de restricciones. Además, en muchos casos, las configuraciones de PC que son más antiguas, de 4 a 6 años, se pueden usar con éxito para esta función.

Los servidores de laboratorio deben tener preferiblemente instalado el sistema operativo Linux. En cuanto a los requisitos de rendimiento, normalmente el sistema operativo Linux tiene demandas más bajas, y las versiones recientes, como Ubuntu 20.04 pueden ejecutarse con éxito en configuraciones de PC más antiguas. EOL (fin de vida útil) para Ubuntu LTS (soporte a largo plazo) se extiende a varios años, por lo que se puede usar en los próximos años.

Si se utiliza el sistema operativo Windows, se recomienda permanecer con la versión más reciente (Win 10 o Win 11), principalmente debido a problemas de seguridad y de soporte de terceros (por ejemplo, controladores). Por lo general, las versiones modernas de Win OS tienen un mayor requisito de rendimiento de PC (en comparación con Linux), pero esto no debería ser una preocupación para las versiones de sistemas operativos instalados por OEM y los PCs que tienen pocos años. La facilidad de uso si se requieren paquetes de software adicionales y la familiaridad con el sistema operativo son factores de decisión mucho más importantes.

Compatibilidad con la red del servidor de laboratorio

El servidor de laboratorio debe estar conectado a la LAN interna del campus o, en casos raros, para una instalación más pequeña, directamente a Internet público, a través de DSL, cable, 4G o módem de fibra óptica.

En el primer caso, la conexión a través de Ethernet de 100/1000 Mbps a la LAN interna del campus, el acceso a Internet público estará normalmente habilitado para el servidor del laboratorio, pero no es obligatorio. Sin embargo, el acceso al servidor 'Bastion' desde la LAN interna sí debe estar habilitado necesariamente.

Un detalle importante definirá muchas acciones de seguimiento. ¿Hay direcciones IP públicas disponibles para los servidores del laboratorio? ¿Queremos exponer completamente los servidores de laboratorio con equipos de laboratorio potencialmente sensibles a la Internet pública? Debemos tener en cuenta que las direcciones IP públicas son recursos escasos y para el esquema de direccionamiento IPv4, solo hay unos pocos miles de millones de direcciones únicas disponibles.

Un buen consejo es evitar tener servidores de laboratorio directamente accesibles en Internet público con sus propias direcciones IP. Si esta sigue siendo la decisión (tal vez temporal), asegúrese de que para Windows los parches de seguridad se apliquen inmediatamente tan pronto como se publiquen. Usar una dirección IP pública tiene el mismo riesgo que la ventaja que ofrece: permite a CUALQUIERA (intentar) conectarse a su dispositivo directamente desde Internet. Dado que la comunicación a través de la Internet pública SIEMPRE es bidireccional, cuando se conecta a Internet, Internet se conecta a usted. Al explotar varias vulnerabilidades, las personas con malas intenciones pueden acceder a sus archivos, modificar los parámetros de acceso o realizar anónimamente cualquier acción que les plazca. Incluso hay servicios de Internet disponibles públicamente, sin mencionar el escáner personalizado, que escanean regularmente todas las direcciones IP en el espacio público y buscan vulnerabilidades conocidas. Además, su dirección IP real se puede utilizar no sólo para hackear su red de laboratorio, sino también como plataforma de lanzamiento para realizar ataques DDoS, en ambas direcciones. En este caso, su red

bombardearía a otros, o externamente su enrutador y red pueden verse abrumados con ataques DDoS. Tales ataques, a menudo se llevan a cabo contra jugadores y streamers, por ejemplo, para eliminar a un oponente de la competencia sabotando su conexión a Internet.

Sin embargo, temporalmente (si es posible), puede establecer la dirección IP pública, para soluciones rápidas y, con mayor frecuencia, para configuraciones iniciales. Por ejemplo, si está haciendo la configuración inicial de un WebLab, y lo hace de forma remota, alguien localmente necesitará primero instalar el sistema operativo y aprovisionar, por ejemplo, en el caso del sistema operativo Linux, acceso SSH, enganchar a la LAN y asignar una dirección IP pública estática. Después de eso, todo lo demás se puede hacer de forma remota, como se describe en detalle en "RELAB_Weblab_IO5_OffLine2OnLine.pdf".

En este último caso, con respecto a la conectividad de red, si el servidor del laboratorio (para alguna configuración aislada), se conecta a través de DSL, cable, fibra, 4G, con dirección IP dinámica asignada, y se encuentra detrás del enrutador NAT, hay una opción para abrir un puerto específico, pero solo si la dirección IP estática está disponible para su módem. En casos más frecuentes, esto no es posible. Aún así, mediante el uso del servidor "Proxy Jump", que se puede aprovisionar de forma remota, podemos hacer que el servidor de laboratorio esté disponible para los estudiantes remotos: el servidor de laboratorio necesitaría iniciar la conexión de túnel SSH inversa (como se describe en "RELAB_Weblab_IO5_OffLine2OnLine.pdf").

El ancho de red requerido para garantizar una buena conectividad de escritorio remoto con cualquiera de los métodos anteriores no es alto, por lo que las conexiones con 2Mbps o más serán suficientes. La baja latencia es más importante, ya que cualquier latencia (ping RTT) más allá de 150 ms (o 200 ms como máximo) hará que la experiencia de conexión de escritorio remoto sea inaceptable.

Complementos de accesibilidad remota de Lab Server

Si se utiliza el sistema operativo Windows, es necesario habilitar el Firewall / Defender de Windows o utilizar un software de terceros, por ejemplo, Norton 360. Después de eso, el acceso a Escritorio remoto de Windows debe estar habilitado. Tenga en cuenta que Escritorio remoto de Windows no está habilitado en las ediciones de Windows Home de forma predeterminada.

Una alternativa de código abierto muy común a este RDP de Windows es el servicio VNC, que se puede instalar en todas las versiones de Windows, así como en el sistema operativo Linux. La mayoría de las versiones gratuitas de VNC no tienen cifrado del tráfico incorporado. Existen varias implementaciones de este servicio, basadas en el protocolo común definido en RFC 6143. El uso compartido de archivos X2GO debe habilitarse explícitamente en el servidor y el cliente, proporcionando carpetas asignadas que se sincronicen de forma segura. En el lado del servidor, la carpeta asignada CLIENT-FOLDER aparecerá como ~/media/disk/_CLIENT-FOLDER ('/' o '\' se reemplazan con '_').

Para un servidor de laboratorio basado en el sistema operativo Linux (no disponible para el servidor del sistema operativo Windows), aparte de la opción de servidor VNC, el paquete de software de código abierto X2GO (<https://wiki.x2go.org/doku.php/download:start>) permite conexiones de baja latencia, basadas en SSH, por lo tanto, una solución totalmente cifrada a través de la red. La conexión del cliente X2GO server<=>X2GO permite compartir archivos. X2GO se basa en el protocolo NX3 modificado, que hace compresión de eventos X11:

```
RemoteClient (e.g. xterm) <=> nxproxyClient <=>(Web)<=> nxproxyServer<=>X server
```

No solo los entornos de escritorio Linux son adecuados para X2GO, sino que XFCE, LXDE y MATE siempre funcionarán. Por lo tanto, es posible que se requiera un entorno de escritorio adicional (como se describe en "RELAB_Weblab_IO5_OffLine2OnLine.pdf").



Co-funded by
the European Union

Tanto VNC (<https://github.com/RealVNC>, <https://github.com/TigerVNC/tigervnc>, <https://github.com/ultravnc/UltraVNC>) como X2GO (https://code.x2go.org/releases/X2GoClient_latest_mswin32-setup.exe) tienen clientes Windows.

Detalles de configuración del servidor público Bastión/Weblab (Linux, Ubuntu)

VPS Linux generalmente ya está configurado para roles públicos con los paquetes necesarios instalados y algún nivel de medidas de seguridad de red aprovisionadas. Este es el caso de soluciones comerciales como AWS, Google, Digital Ocean, Linode, Azure o servicios en la nube alojados en la universidad. Esto significa que el acceso SSH ya está habilitado, con pocas cuentas creadas, al menos una de ellas con privilegios 'sudo-er' para ejecutar pasos de seguimiento. Puede residir en la misma LAN o en varias LAN incluso detrás de diferentes firewalls.

Un solo servidor bastión es suficiente para varios equipos de laboratorio y está limitado por el ancho de banda disponible y la potencia de cómputo en menor medida. Con esta configuración, Bastion Server será el único servidor Linux que acepte conexiones SSH públicas.

Si un usuario desea acceder a la máquina de control de un laboratorio "oculta" (desde Internet pública), primero debe conectarse al bastión y realizar otra conexión SSH desde el bastión hasta el destino final. Este proceso a veces se denomina "salto de proxy" y se puede automatizar.

Los detalles sobre cómo configurar el servidor Bastion se proporcionan en "RELAB_Weblab_IO5_OffLine2OnLine.pdf"

Configuración de la estación de trabajo cliente / estudiante (Windows o Linux)

Para habilitar la experiencia de Escritorio remoto, usaremos el paquete de software X2GO, instalando el lado del servidor del laboratorio y el lado del cliente X2GO en la estación de trabajo del estudiante. El lado del servidor, X2GO solo está disponible para Linux.

El enfoque actual se basa en la distribución estática de los estudiantes entre los grupos a los que se les asigna una sola configuración de laboratorio (del mismo tipo) por grupo. Como seguimiento, proporcionaremos un servicio de reserva para que los estudiantes puedan reservar un horario en una configuración. Al implementar un servicio personalizado de agente de sesión X2GO, también proporcionaremos la capacidad de enrutamiento a la configuración disponible del tipo seleccionado. De esta manera, el rol del estudiante se simplifica aún más, por lo que las claves SSH y otros parámetros de sesión X2GO se intercambiarán dinámicamente al comienzo de la sesión X2GO.

Los enfoques adicionales, como uno basado en el software Apache Guacamole (<https://guacamole.apache.org/>), pueden permitir un verdadero enfoque "sin cliente", es decir, sin ningún paquete de software adicional instalado en el lado del PC del estudiante.

Este enfoque se limita a servidores VNC y servidores RDP gratuitos. En este caso, el estudiante solo tendría que apuntar al servicio Guacamole (iniciado en el servidor Bastión) y proporcionar credenciales (nombre de usuario y contraseña). Después de eso, se dirigiría al equipo de laboratorio enrutado. Este enfoque también se describe como extensión y alternativa para los servidores Windows hacia el final del proyecto RELAB.