

Avance xd

Avance hasta el momento

1. Solución del problema de la llegada al portal cautivo:

El flow que se creaba que hacía posible la llegada del host (h1 o h2) hacia el servidor del portal cautivo vencía luego de terminado el idle timeout, lo cual es un comportamiento esperado pero el problema venía en que luego ya no había forma de volver a instalar el flow pues debido a un etiquetado en el código del módulo del PacketInManager "CAPTURED_TO_PORTAL" el controlador se negaba a reinstalar los flows entonces se corrige y solo se evitará reinstalar cuando el estado sea FULL_ACCESS (hay que definir mejor esto).

Resultado: Cada que se desea acceder al servidor del portal cautivo se logra incluso cuando el idle time vence pues se reinstalan las reglas.

2. Acerca de DHCP:

Se manejan mediante PacketIn y PacketOut, mediante FLOOD, es por eso que no vemos reglas instaladas para este, este comportamiento tiene sentido puesto que DHCP funciona por broadcast, es un diseño recomendado.

3. Se instala interfaz gráfica para H1 y además se habilita todo su almacenamiento:

- Con respecto al tema del almacenamiento si introducimos:

```
lsblk -o NAME,SIZE,TYPE,MOUNTPOINT
```

Veremos de salida el espacio con el que cuenta el host, notaremos que no se está utilizando los 15 GB completos entonces podemos pasárselo hacia vda1 que según entiendo es la partición donde se instala todo lo que descarguemos.

Entonces extendemos:

```
sudo growpart /dev/vda 1
sudo resize2fs /dev/vda1
```

De esta forma los 15 GB son utilizados por completo, esto será importante para instalar la interfaz gráfica.

- Se instala entorno gráfico liviano:

```
sudo apt update  
sudo apt install xfce4 xfce4-goodies -y  
sudo apt install lightdm -y  
sudo systemctl enable lightdm --now  
sudo reboot
```

- Se instala vncserver con el objetivo de abrir una sesión gráfica:

```
sudo apt install tightvncserver -y
```

- Creamos la sesión gráfica:

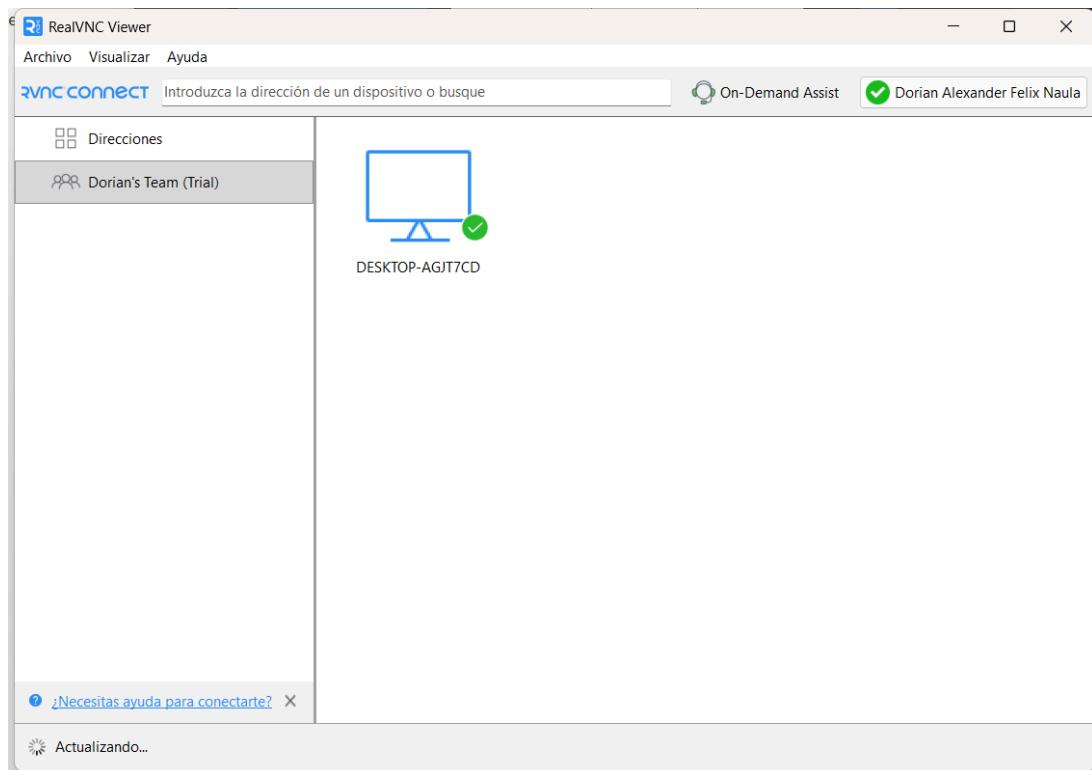
```
vncserver :1
```

- Se decide instalar el navegador chromium (es el único que me dejó instalar xd)

```
sudo apt update  
sudo apt install chromium-browser
```

TENER EN CUENTA QUE SE EJECUTA EN EL PUERTO 5811

- Ahora ¿Cómo acceder a la interfaz gráfica?, se utiliza un cliente VNC, yo utilice RealVNC, lo descargan y una vez dentro tendrán lo siguiente:



En el buscador ingresan la dirección de la vm correspondiente, pero en este caso debemos tener en cuenta que nosotros no nos conectamos por la dirección ip de la vm sino por la dirección ip del gateway y un puerto correspondiente entonces antes debemos realizar lo siguiente:

1. Creamos un túnel SSH desde nuestra laptop hasta la VM:

```
ssh -p 5811 -L 5901:localhost:5901 ubuntu@10.20.12.33
```

2. Hay que asegurarnos también de lo siguiente:

```
mkdir -p ~/.vnc  
nano ~/.vnc/xstartup
```

Esto corresponde a la creación de un correcto archivo de configuración, entonces pegamos esto:

```
#!/bin/bash  
xrdb $HOME/.Xresources  
startxfce4 &
```

y lo hacemos ejecutables:

```
chmod +x ~/.vnc/xstartup
```

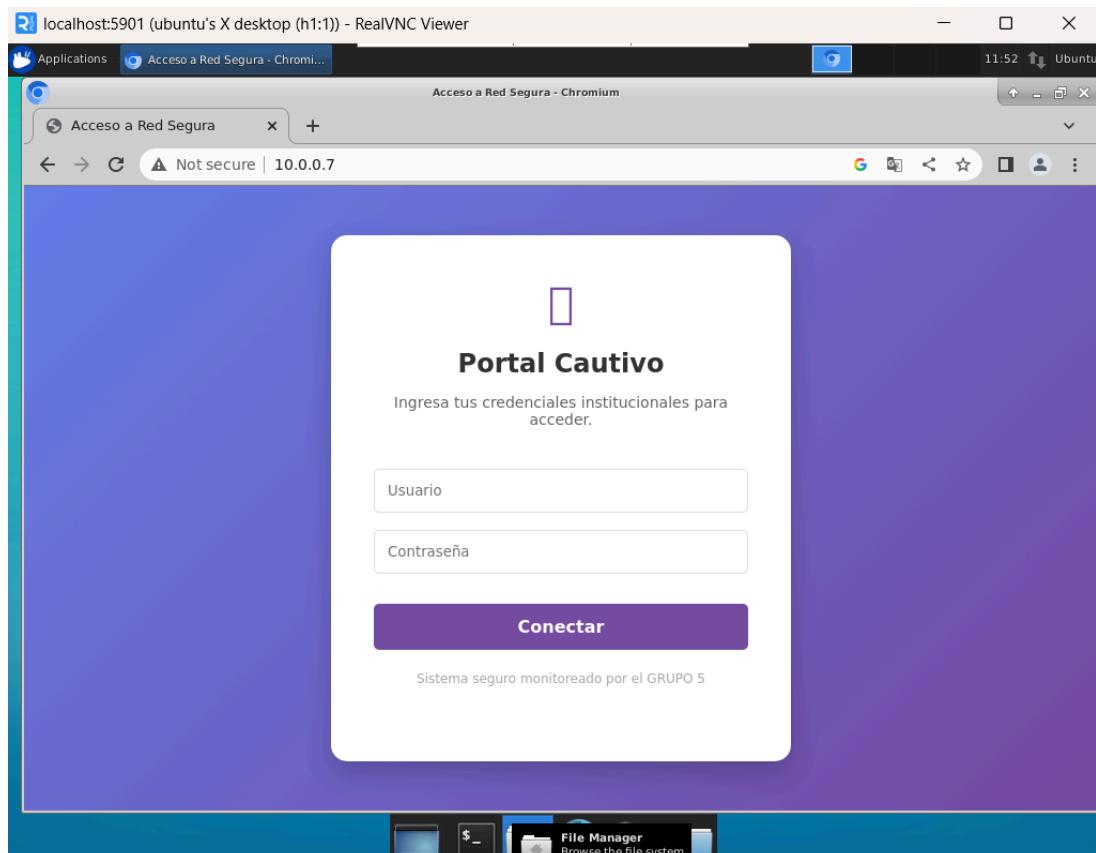
Iniciamos de nuevo el servidor (matamos antes cualquier sesión vieja):

```
vncserver :1
```

3. Ahora debemos mantener abierta la consola donde nos hemos conectado por ssh y procedemos a introducir lo siguiente en el VNC viewer:

```
localhost:5901  
o también  
localhost:1
```

Entonces accederemos a la GUI y abrimos el navegador y ya podemos probar el portal cautivo xd



4. Separación de base datos, para identificación (por defecto de RADIUS) y para administrar info de usuarios, roles, cursos, recursos:

Se borró la tabla de “user_roles” de la bd que se utiliza por defecto con RADIUS, esto pues se implementó la bd que se propuso, con ciertas modificaciones, entonces se borra la tabla mencionada y se levanta la nueva bd.

- La nueva bd modificada es:

```
-- =====
=====
-- BASE DE DATOS SDN_POLICY
-- Control Académico + Autorización SDN
-- Integración Portal Cautivo + Floodlight
-- =====
=====

DROP DATABASE IF EXISTS sdn_policy;
CREATE DATABASE sdn_policy DEFAULT CHARACTER SET utf8mb
4;
USE sdn_policy;

SET FOREIGN_KEY_CHECKS = 0;

-- =====
=====
-- 1. Tabla careers (carreras)
-- =====
=====

CREATE TABLE careers (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL UNIQUE
) ENGINE=InnoDB;

-- =====
=====
-- 2. Tabla users (usuarios académicos)
-- =====
```

```

=====
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(100) NOT NULL UNIQUE, -- coincide con
RADIUS username
    fullname VARCHAR(100),
    lastname VARCHAR(100),
    email VARCHAR(100) UNIQUE,
    career_id INT,
    FOREIGN KEY (career_id) REFERENCES careers(id)
        ON DELETE SET NULL
        ON UPDATE CASCADE
) ENGINE=InnoDB;

-- =====
=====

-- 3. Tabla roles (roles globales del sistema)
-- =====
=====

CREATE TABLE roles (
    id INT AUTO_INCREMENT PRIMARY KEY,
    role_name VARCHAR(50) NOT NULL UNIQUE
) ENGINE=InnoDB;

-- =====
=====

-- 4. Relación users ↔ roles
-- =====
=====

CREATE TABLE user_roles (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    role_id INT NOT NULL,
    UNIQUE(user_id, role_id),
    FOREIGN KEY (user_id) REFERENCES users(id)

```

```

        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (role_id) REFERENCES roles(id)
        ON DELETE CASCADE
        ON UPDATE CASCADE
) ENGINE=InnoDB;

-- =====
=====

-- 5. Tabla courses (cursos)
-- =====
=====

CREATE TABLE courses (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    code VARCHAR(20) NOT NULL UNIQUE,
    career_id INT,
    FOREIGN KEY (career_id) REFERENCES careers(id)
        ON DELETE SET NULL
        ON UPDATE CASCADE
) ENGINE=InnoDB;

-- =====
=====

-- 6. Relación users ↔ courses
-- =====
=====

CREATE TABLE user_courses (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    course_id INT NOT NULL,
    role_in_course ENUM('alumno','profesor') NOT NULL,
    UNIQUE(user_id, course_id),
    FOREIGN KEY(user_id) REFERENCES users(id)
        ON DELETE CASCADE
        ON UPDATE CASCADE,

```

```

FOREIGN KEY(course_id) REFERENCES courses(id)
    ON DELETE CASCADE
    ON UPDATE CASCADE
) ENGINE=InnoDB;

-- =====
=====

-- 7. Tabla resources (recursos reales accesibles)
-- =====
=====

CREATE TABLE resources (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    server_ip VARCHAR(15) NOT NULL,
    server_port INT NOT NULL DEFAULT 80,
    protocol ENUM('tcp', 'udp', 'any') DEFAULT 'tcp',
    critical BOOLEAN DEFAULT 0,
    role_required INT NULL,      -- si un recurso requiere rol específico
co
    FOREIGN KEY (role_required) REFERENCES roles(id)
        ON DELETE SET NULL
        ON UPDATE CASCADE
) ENGINE=InnoDB;

-- =====
=====

-- 8. Relación courses ↔ resources (M:N)
-- =====
=====

CREATE TABLE course_resources (
    id INT AUTO_INCREMENT PRIMARY KEY,
    course_id INT NOT NULL,
    resource_id INT NOT NULL,
    UNIQUE(course_id, resource_id),
    FOREIGN KEY(course_id) REFERENCES courses(id)
        ON DELETE CASCADE

```

```
    ON UPDATE CASCADE,
    FOREIGN KEY(resource_id) REFERENCES resources(id)
        ON DELETE CASCADE
        ON UPDATE CASCADE
) ENGINE=InnoDB;
```

```
-- =====
=====
-- 9. Recursos adicionales por usuario (excepciones)
-- =====
=====
```

```
CREATE TABLE user_resources (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    resource_id INT NOT NULL,
    UNIQUE(user_id, resource_id),
    FOREIGN KEY(user_id) REFERENCES users(id)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY(resource_id) REFERENCES resources(id)
        ON DELETE CASCADE
        ON UPDATE CASCADE
) ENGINE=InnoDB;
```

```
SET FOREIGN_KEY_CHECKS = 1;
```

- En h7 se levantó la bd con el nombre de sdn_policy.sql, el usuario asignado a la bd es **dti_dodo** y contraseña **ubuntu**.
- Para ingresar a la bd:

```
mysql -u dti_dodo -pubuntu sdn_policy
```

- Se introdujeron los roles:

```
INSERT INTO roles (role_name) VALUES
('ROLE_ADMIN'),
```

```
('ROLE_PROFESSOR'),  
('ROLE_RESEARCHER'),  
('ROLE_STUDENT'),  
('ROLE_GUEST');
```

- Las carreras:

```
INSERT INTO careers (name) VALUES  
('Ingeniería de Telecomunicaciones'),  
('Ingeniería Informática'),  
('Ingeniería civil');
```

- Usuarios de prueba:

```
INSERT INTO users (username, fullname, lastname, email, career_id) VALUES  
('admin@uni.edu', 'Admin', 'Master', 'admin@uni.edu', 1),  
('prof@uni.edu', 'Carlos', 'Profesor', 'prof@uni.edu', 1),  
('alumno@uni.edu', 'Juan', 'Alumno', 'alumno@uni.edu', 1);
```

- Se asignan los roles:

```
INSERT INTO user_roles (user_id, role_id) VALUES  
(1, 1), -- admin → ROLE_ADMIN  
(2, 2), -- prof → ROLE_PROFESSOR  
(3, 4); -- alumno → ROLE_STUDENT
```

- Los cursos:

```
INSERT INTO courses (name, code, career_id) VALUES  
('SDN', 'TEL354', 1),  
('Ingenieria Inalambrica', 'TEL280', 2);
```

- Se inscriben profesor y alumno a curso:

```
-- Profesor  
INSERT INTO user_courses (user_id, course_id, role_in_course) VALUES
```

```
(2, 1, 'profesor');

-- Alumno matriculado
INSERT INTO user_courses (user_id, course_id, role_in_course) VALUES
(3, 1, 'alumno');
```

- OJO: Aún no se ha hecho el llenado de la tabla de recursos.
- Por último se llena la bd de Radius con las credenciales.

```
mysql -u radius -pubuntu radius
```

```
USE radius;
```

```
INSERT INTO radcheck (username, attribute, op, value) VALUES
('admin@uni.edu', 'SHA-Password', ':=', SHA1('admin123')),
('prof@uni.edu', 'SHA-Password', ':=', SHA1('prof123')),
('alumno@uni.edu', 'SHA-Password', ':=', SHA1('alumno123'));
```

5. Se corrige y adapta el código del portal cautivo según los cambios realizados en la BD.

Hasta el momento todo es funcional, quedan ciertos puntos que corregir y retocar pero hasta el momento el flujo hasta el portal cautivo y la autorización funciona, además super admin puede crear nuevos usuarios (también falta corregir los datos que ingresa).

- Código de app.py (está en h7 xd o lo pasé por wsp)