Ciclo formativo Grado Superior a distancia:

ASIR

[Institución]

[Campus]

Diseño e Implementación de una Casa Inteligente Segmentada con pfSense y Home Assistant

Tutora de Proyecto: [Nombre de tutor/a]

Autor/a: [Tu nombre y apellidos]

Roberto Montoro García

Beatriz Rodríguez Ruíz de Pascual

Curso académico: 2024/2025

Contenido

Nota (versión pública): Este documento ha sido saneado para publicación. Se han eliminado o anonimizado datos personales y credenciales. Las contraseñas y secretos se reemplazan por marcadores como CHANGEME_STRONG_PASSWORD.

1. Introducción

1.1. Contexto del proyecto

La domótica ha revolucionado la forma en que se gestionan los hogares, permitiendo controlar los dispositivos de forma remota y automática a través de redes locales o acceso a Internet. Sin embargo, este cambio de paradigma también ha generado importantes desafíos en términos de seguridad, especialmente con la proliferación de dispositivos IoT en infraestructuras domésticas y conexiones de red. El aumento del número de vulnerabilidades que se añaden al facilitar la creación de dispositivos conectables hace que sea absolutamente necesario utilizar soluciones que permitan una red segmentada y controlada. A nivel técnico, se propone el uso de pfSense como herramienta profesional para la gestión avanzada de redes y seguridad, así como Home Assistant para la orquestación centralizada de todos los elementos de domótica.

Este proyecto forma parte del Grado Superior en Administración de Sistemas Informáticos en Red, y su principal objetivo es simular e implementar una infraestructura doméstica segura, replicable con herramientas profesionales en un entorno real.

1.2. Motivación y justificación

La razón principal para emprender este proyecto es la necesidad real de soluciones de automatización doméstica seguras en los hogares. A medida que los asistentes virtuales, las cerraduras inteligentes, los sensores de movimiento, las cámaras y otros dispositivos conectados entran en nuestros hogares, la exposición a los riesgos de ciberseguridad es significativamente mayor. Muchas de las soluciones comerciales actuales no permiten la segmentación de la red ni la creación de reglas de control específicas. Herramientas

como pfSense nos permiten implementar arquitecturas avanzadas incluso en despliegues domésticos, pero con funcionalidades de red empresarial.

Además, la elección de Home Assistant como programa central del proyecto se debe a su flexibilidad, su capacidad para trabajar con diversos protocolos y dispositivos, y la accesibilidad de su fuente, lo que permite considerarlo no solo como una solución funcional, sino también educativa. Esto se explica por el hecho de que la implementación de la solución permite no solo el uso de los conocimientos aprendidos durante la formación teórica, sino también la propuesta de una solución real que puede adaptarse para su uso en la vida real.

1.3. Alcance del proyecto

Este trabajo implica la simulación e implementación de una red doméstica inteligente, en particular la selección de tres VLAN para dispositivos IoT e invitados segregados. Esta tarea se implementa en un entorno virtualizado en máquinas virtuales con VirtualBox; para la seguridad y el enrutamiento ha sido necesario integrar pfSense. La orquestación de dispositivos y automatizaciones se realiza mediante Home Assistant.

Los objetivos incluyen:

- Formateo lógico del proyecto en Cisco Packet Tracer.
- La creación y configuración de máquinas virtuales para la automatización y el control.
- La migración de redes DHCP.
- Funcionamiento basado en pruebas de acceso, segregación y automatización.

Cabe señalar que la implementación física final o intranet en línea se llevará a cabo con fines educativos y de demostración. El presente trabajo se estructura en nueve capítulos que abarcan desde el marco teórico y la justificación técnica, hasta el desarrollo práctico, los resultados, las conclusiones y los anexos. Además de la segmentación de red y

automatización básica, se ha integrado OpenVPN para acceso remoto seguro, pfBlockerNG para el bloqueo de amenazas y una bombilla física con firmware Tasmota para validar la interacción con dispositivos reales. En conjunto, se busca no solo demostrar la viabilidad del entorno propuesto, sino también documentar de forma detallada su aplicación educativa y técnica mediante ejemplos y evidencias prácticas.

2. Hipótesis

2.1. Planteamiento de la hipótesis

La hipótesis principal de este proyecto se basa en la necesidad real de la aplicación: la segmentación de redes equipadas con VLAN y conectadas a pfSense y la automatización empleada por Home Assistant permiten una mayor seguridad, control y comodidad en el uso de sistemas domésticos automatizados.

En la mayoría de los hogares actuales, todos los dispositivos comparten la misma red y, por lo tanto, la misma amenaza de seguridad. Esto significa que, si un dispositivo de la red se ve comprometido, el atacante puede acceder a todos los demás elementos críticos de la red. Por lo tanto, se postula que con una arquitectura segmentada es posible evitar la propagación de riesgos. En cuanto a la suposición relativa al uso de herramientas de código abierto, se puede argumentar que no solo es factible en entornos de laboratorio o educativos, sino que también es fácilmente replicable en la vida real a bajo coste.

2.2. Implicaciones de la hipótesis

Si la hipótesis es correcta, esto llevaría a que cualquier usuario con conocimientos básicos de infraestructura y sistemas de red podría llevar a cabo los siguientes procesos: aplicar segmentación lógica en su red doméstica sin necesidad de costosos equipos ni conocimientos especializados, proteger sus dispositivos IoT en la red contra amenazas internas y externas mediante reglas de cortafuegos cuidadosamente definidas, gestionar su hogar automatizado desde un estado seguro y centralizado con cualquier dispositivo

inteligente reemplazable y adaptable, aprovechar un ejemplo escolar real en la enseñanza técnica de redes, seguridad y automatización.

Por lo tanto, este proyecto intenta validar que una arquitectura segmentada mediante VLANs, con herramientas de software libre como pfSense y Home Assistant, puede implementarse en entornos educativos con fines formativos, pero también replicarse en instalaciones reales de forma segura, económica y funcional.

3. Objetivos

3.1. Objetivo general

Diseñar e implementar una infraestructura de hogar inteligente segmentada y segura, que integre dispositivos IoT, usuarios invitados y servicios administrativos, utilizando pfSense para el control de red y Home Assistant como sistema de automatización.

3.2. Objetivos específicos

Se propone validar que mediante el uso de herramientas de software libre y entornos virtualizados es posible construir un sistema de domótica eficiente, seguro y escalable, replicable en situaciones reales y con un enfoque educativo. Y para cumplir los objetivos mencionados, se enumeran los siguientes objetivos específicos:

- Configurar un entorno virtualizado en VirtualBox para simular la infraestructura de red de un hogar inteligente.
- Instalar y segmentar redes con pfSense, con tres VLAN claramente diferenciadas.
- Asignar direccionamiento IP automático y definir reglas de firewall para el aislamiento entre VLANs.
- Integrar Home Assistant en la red de administración y permitir su comunicación controlada con dispositivos IoT.

- Simular la red lógica en Cisco Packet Tracer con dispositivos reales.
- Documentar y validar el entorno con pruebas de conectividad, automatización y aislamiento.

4. Marco teórico

4.1. Domótica: concepto y evolución

La domótica se define como el conjunto de tecnologías aplicadas al control inteligente y la automatización de una vivienda, mediante el control eficiente de sus redes de energía, seguridad, confort y comunicaciones. Se ha desarrollado gracias a la evolución de los sensores, los actuadores, las tecnologías de redes inalámbricas y las plataformas de integración de sistemas como Home Assistant.

En sus inicios, los sistemas de domótica eran sistemas cerrados y caros, solo accesibles a proyectos de alto coste. Con la proliferación del Internet de las cosas (IoT) y la apertura de soluciones de código abierto, la domótica se ha vuelto accesible para el público en general.

4.2. Seguridad en entornos domóticos

Uno de los principales retos de la domótica es la seguridad de la información. Los dispositivos IoT suelen estar conectados a Internet, no se actualizan con frecuencia y se configuran de forma insegura por defecto. En consecuencia, estos dispositivos son blancos fáciles de ataques.

La segmentación de la red, el uso de cortafuegos, el software actualizado y la gestión de accesos son esenciales para preservar la integridad y la privacidad del usuario. De lo contrario, una arquitectura domótica insegura puede poner en peligro toda la red doméstica.

4.3. Arquitectura de redes LAN y segmentación

Una LAN, red de área local, es una red que conecta múltiples dispositivos en un espacio confinado. En un entorno domótico, una arquitectura LAN permite que los sensores, las cámaras, los asistentes virtuales, los teléfonos inteligentes y los servidores se comuniquen.

4.3.1. VLANs y su funcionalidad

Las VLAN, redes de área local virtuales, permiten crear múltiples dominios lógicos aislados dentro de una red física. Cada una de estas VLAN actúa prácticamente como una red física independiente a pesar de utilizar la misma infraestructura física.

Por ejemplo, podría tener una VLAN10 para los dispositivos IoT y una VLAN20 para los invitados. De esta manera, puede utilizar diferentes reglas de seguridad y supervisar el tráfico entre ellas.

4.3.2. Seguridad en redes segmentadas

El uso de VLANs facilita:

- La limitación de difusión de tráfico.
- Separar dispositivos con diferentes perfiles de riesgo.
- Implementar de políticas de seguridad distintas por segmento.
- Evitar accesos no autorizados a recursos críticos.

El firewall pfSense permite hacer visible esta arquitectura permitiendo, bloqueando y filtrando el tráfico en consecuencia

4.4. pfSense como solución de firewall

4.4.1. Características principales

PfSense es un sistema operativo FreeBSD utilizado para gestionar y administrar un cortafuegos y un router de red. Su Community Edition es gratuita, y es potente y muy utilizada en la industria y la educación.

Entre sus principales características destacan:

- VLAN y subredes gestionables. Configuración avanzada de reglas de cortafuegos.
- Los cortafuegos también son compatibles con VPN, balanceo de carga e IDS.
- Interfaz web para la administración y la capacidad innata de migrar a la versión premium
- Comunidad y documentación.

4.4.2. Comparativa con otros firewalls

En comparación con otros cortafuegos, comparado con un hogar, pfSense ofrece más control, personalización y seguridad. OpenWRT o DD-WRT no parecían funcionar para ese caso, ya que se trata de dispositivos de alto perfil para usuarios técnicos.

4.5. Home Assistant como sistema de automatización

4.5.1. ¿Qué es Home Assistant?

Home Assistant es una plataforma de domótica de código abierto que permite controlar, supervisar y automatizar dispositivos IoT con un único sistema operativo que se ejecuta localmente, por lo que no hay que preocuparse por la seguridad de la información.

Se integra con más de 1000 dispositivos y servicios como luces inteligentes, sensores, asistentes de voz (Google, Alexa), cámaras IP, enchufes, alarmas, etc. Recientemente, se integró Home Assistant en una VLAN específica, lo que facilitó la gestión del tráfico y aumentó la seguridad al permitir una comunicación controlada solo con dispositivos IoT.

4.5.2. Integración con redes segmentadas

Home Assistant puede integrarse fácilmente en una arquitectura de red segmentada, como la que se ha implementado en este proyecto. En este caso, se ha ubicado dentro de la VLAN1 (red de administración), ya que actúa como núcleo del sistema domótico y debe estar protegido del tráfico de dispositivos IoT y de invitados. Esto le permite interactuar

de forma segura con sensores y actuadores sin exponer otros servicios internos a potenciales riesgos.

Gracias a esta segmentación, Home Assistant, ubicado en la red de administración (VLAN1), solo es accesible desde la VLAN IoT mediante reglas específicas de firewall, lo que permite su comunicación con dispositivos autorizados sin exponer el resto de la red, mientras que el acceso externo está controlado por las políticas de seguridad definidas en pfSense.

Además, esta arquitectura permite ampliar las funcionalidades del sistema domótico integrando protocolos específicos como MQTT y utilizado el broker Mosquitto, ejecutado en una máquina Linux dentro de la VLAN IoT, para simular sensores que publican datos a Home Assistant. Esta integración refuerza la automatización del entorno domótico, permitiendo crear reglas reactivas y notificaciones inteligentes en función de los eventos detectados en la red.

4.6. Protocolo MQTT y broker Mosquitto en domótica

MQTT (Message Queuing Telemetry Transport) es un protocolo de mensajería ligero y eficiente, muy utilizado en entornos de automatización y dispositivos IoT. Ideal para comunicaciones máquina a máquina (M2M) en entornos con ancho de banda reducido, como redes domésticas con sensores IoT. Funciona bajo un modelo publish/subscribe (publicación/suscripción), en el que los dispositivos se comunican a través de temas (topics) gestionados por un servidor intermedio llamado broker.

Uno de los brokers MQTT más populares es Mosquitto, una solución de código abierto, ligera y compatible con sistemas Linux y Windows. Mosquitto actúa como intermediario entre los sensores, actuadores y plataformas de control, permitiendo que todos intercambien datos de forma eficiente. En este proyecto se ha utilizado como nexo entre los dispositivos IoT y Home Assistant.

5. Material y métodos

Este apartado describe los recursos utilizados y los procedimientos seguidos para implementar una red domótica segmentada en un entorno virtualizado. La metodología parte de una aproximación práctica y progresiva: desde la creación de máquinas virtuales hasta la simulación topológica en Cisco Packet Tracer.

5.1. Recursos software utilizados

5.1.1. VirtualBox

VirtualBox VirtualBox es un software de virtualización de código abierto que permite ejecutar sistemas operativos invitados en un equipo host. Es un instrumento para configurar entornos independientes con flexibilidad según el caso.

5.1.2. pfSense

pfSense es un cortafuegos y enrutador basado en FreeBSD, y se utilizaron instancias del mismo como método de segmentación al implementar VLAN, servidores DHCP independientes y reglas de cortafuegos entre redes.

5.1.3. Home Assistant

Home Assistant es una plataforma de automatización del hogar que se ejecutaba en una máquina virtual, sobre una máquina VirtualBox, encargándose de controlar todos los dispositivos simulados que existen en la red doméstica virtual.

5.1.4. Cisco Packet Tracer

Cisco Packet Tracer es un simulador de red que permite a los usuarios generar y configurar redes reales con posibles topologías y productos Cisco. Se utilizó para generar la topología que conforma la red segmentada, de tal manera que antes de configurar la red en los entornos de virtualización, se simuló completando el comportamiento lógico.

5.2. Infraestructura de red planteada

La propuesta de infraestructura de red se divide en tres segmentos independientes:

- VLAN 1 (Administración), para la red de administración, donde se encuentran los servicios internos como la gestión de pfSense, y la plataforma Home Assistant.
- VLAN 10 (IoT), para los dispositivos domésticos inteligentes del usuario, tanto sensores como actuadores, con una máquina virtual Ubuntu que aloja el broker MQTT Mosquitto, el cual se encarga de recibir y redirigir mensajes entre los sensores/actuadores conectados y la plataforma domótica Home Assistant.
- VLAN 20 (Invitados), para usuarios externos, como dispositivos móviles con acceso limitado.

Estas VLAN están gestionadas por pfSense, que actúa como puerta de enlace y cortafuegos entre estas tres redes.

5.3. Entorno virtualizado: configuración de máquinas

Máquina Virtual - pfSense:

- Sistema operativo: FreeBSD 64-bit
- RAM: 2 GB
- CPU: 2 núcleos
- Disco duro: 20 GB (dinámico)
- Red:
    - Adaptador 1: NAT (conexión a Internet)
    - Adaptador 2: Red interna (TRUNK)

Máquina Virtual - Home Assistant:

- Sistema operativo: Linux (Other 64-bit)
- RAM: 3 GB

- CPU: 2 núcleos
- Disco duro: 32 GB (formato VDI)
- Red:
    - Adaptador 1: Red interna (TRUNK)

Máquina Virtual - Ubuntu:

- Sistema operativo: Ubuntu 64-bit
- RAM: 2 GB
- CPU: 1 núcleos
- Disco duro: 25 GB (formato VDI)
- Red:
    - Adaptador 1: Red interna (TRUNK)

5.4. Asignación de VLANs y direccionamiento

Se asignaron las siguientes redes privadas a cada segmento, gestionadas desde pfSense con direccionamiento automático por DHCP:

- VLAN1 (Administración): 192.168.1.0/24
- VLAN10_IOT: 192.168.10.0/24
- VLAN20_INV: 192.168.20.0/24

Esta segmentación lógica permitió controlar el acceso a Internet, restringir la comunicación entre redes y aplicar reglas de seguridad personalizadas. La configuración técnica completa de subredes y DHCP se encuentra en el punto 6.1.2.

6. Desarrollo del proyecto

Se describirá cómo se implementó paso a paso la infraestructura de red segmentada, desde la configuración de pfSense y Home Assistant hasta una simulación lógica en Cisco Packet Tracer. Se aplicó un enfoque práctico, utilizando las habilidades adquiridas en el curso ASIR.

6.1. Fase 1: Instalación y configuración de pfSense

6.1.1. Creación de la máquina virtual

Se creó una máquina virtual en VirtualBox. La creación de la máquina virtual para pfSense se detalló previamente en el punto 5.3.

Durante el proceso de instalación de pfSense se asignaron las siguientes interfaces de red:

- Em0 como interfaz WAN (Internet).
- Em1 como interfaz LAN (LAN - red de administración) con dirección IP 192.168.1.1/24.

Sobre la interfaz em1 se configuraron después subinterfaces virtuales para transportar tráfico de VLANs (modo TRUNK):

- Em1.10 como interfaz OPT1 (VLAN10_IOT), con dirección IP 192.168.10.1/24.
- Em1.20 como interfaz OPT2 (VLAN20_INV), con dirección IP 192.168.20.1/24

Este diseño permite aislar el tráfico de dispositivos IoT y de usuarios invitados, al mismo tiempo que se mantiene una red dedicada para la administración del sistema y servicios internos. Esto facilita la aplicación de reglas de firewall específicas, mejorando la seguridad y la gestión del tráfico.

La configuración de las subinterfaces VLAN en pfSense puede consultarse en el Anexo 9.2.

6.1.2. Configuración de subredes y DHCP

Inicialmente, intentamos migrar el servicio de DHCP integrado de pfSense a Kea DHCP, una solución más moderna que mejora rendimiento y modularidad, pero finalmente debido a conflictos y limitaciones en entornos virtualizados, especialmente al ejecutarlo en VirtualBox, no fue posible su implementación. Por eso, optamos por utilizar el servidor DHCP integrado clásico, que ofrece mayor estabilidad y compatibilidad en el entorno de pruebas.

Se configuraron tres servidores DHCP independientes:

- LAN (Administración): Rango 192.168.1.100–192.168.1.150
- VLAN10_IOT: Rango 192.168.10.100–192.168.10.150
- VLAN20_INV: Rango 192.168.20.100–192.168.20.150

Esto permitió el direccionamiento automático de dispositivos conectados en cada segmento. Además, se configuró una reserva DHCP estática para el dispositivo Home Assistant, ubicado en la red LAN (Administración), con los siguientes parámetros:

- MAC Address: 08:00:27:2a:67:66
- IP asignada: 192.168.1.10
- Hostname: HomeAssistant

6.1.3. Reglas del firewall

Las reglas de firewall configuradas en pfSense permiten segmentar las redes virtuales y evitar accesos no autorizados entre dispositivos de distintas VLANs.

Cada interfaz de red en pfSense (LAN, VLAN10_IOT, VLAN20_INV) cuenta con su propio conjunto de reglas. Estas reglas determinan qué tipo de tráfico está permitido o denegado en función del origen, destino, protocolo y puerto utilizado.

En la interfaz VLAN10_IOT se permitió el acceso al gateway y al servidor de Home Assistant, pero se bloqueó el tráfico hacia la red de administración y la red de invitados.

En la interfaz VLAN20_INV se permitió el acceso a Internet, pero se bloqueó el acceso hacia las otras redes internas (la red de administración y la red de IOT).

El detalle técnico puede consultarse en el Anexo 9.2.

6.1.4. Copia de seguridad de la configuración en pfSense

Antes de hacer cualquier cambio importante, conviene guardar una copia de la configuración para poder volver al punto de partida por si algo falla. Para ello, en el menú superior de pfSense ve a Diagnostics - Backup & Restore (en algunas versiones aparece bajo System - Configuration - Backups/Restore).

Una vez dentro de la pestaña Backup, simplemente:

- Seleccionar la opción All, para incluir todo el sistema.
- Marcar la casilla Include RRD data, para conservar las estadísticas históricas.
- Finalmente, clicar en Download configuration as XML.

Automáticamente se descargará un archivo XML con toda la configuración del firewall. Guardándolo en un lugar seguro (por ejemplo, en la carpeta del proyecto o en un repositorio) para poder restaurarlo más adelante si hace falta.

6.1.5 Servicios adicionales en pfSense

Además de la segmentación por VLANs, la asignación de IPs y la configuración del firewall, pfSense ofrece otros servicios fundamentales que fueron activados en este proyecto para garantizar la funcionalidad completa de la red.

Entre ellos se configuraron:

- NAT (Network Address Translation), para permitir el acceso a Internet desde las redes internas.
- DNS Resolver, para resolver nombres de dominio tanto locales como externos.

- El portal de gestión web (WebConfigurator), accesible desde la red de administración, con autenticación segura.

La configuración técnica de estos servicios puede consultarse en el Anexo 9.2.

6.2. Fase 2: Instalación y configuración de OpenVPN en pfSense

Para proporcionar acceso remoto seguro a la red implementamos OpenVPN en pfSense. Se configuró una Autoridad Certificadora (CA) y se generó un certificado de servidor "OpenVPN-Server" firmado por la CA "pfSense-CA".

El servidor OpenVPN se ha configurado para escuchar en UDP puerto 1194 en la interfaz WAN, con un túnel seguro (10.10.20.0/24) y autenticación SSL/TLS más usuario. Añadimos las reglas necesarias en WAN y tun interface para permitir el tráfico.

Un usuario fue creado y se exportó su perfil de cliente mediante la herramienta OpenVPN Export. En una máquina virtual Lubuntu se instaló el cliente OpenVPN versión 2.4.12, se modificó el fichero client.ovpn para usar cipher AES-256-CBC y se inició la conexión.

Se verificó la conexión mediante:

- Levantamiento de la interfaz tun0 con IP 10.10.20.2/24.
- Pruebas de conectividad (ping a 10.10.20.1 y a la IP LAN de pfSense 192.168.1.1).
- Confirmación de puerto UDP 1194 accesible.
- Verificación de logs TLS handshake en pfSense y cliente.

El túnel VPN permitió el acceso a la red interna y a Internet desde la máquina cliente.

6.3. Fase 3: Implementación de seguridad adicional con pfBlockerNG

Con el objetivo de proteger los dispositivos IoT y mitigar amenazas externas, se instaló pfBlockerNG en pfSense.

Se activaron:

- IP Feeds (PRI1: EmergingThreats, Spamhaus, CINS Army) con actualización horaria y logging.
- DNSBL para bloquear dominios maliciosos (categorías Ads, Malware, Tracking).

Configuraciones realizadas:

- Ajuste del DNS Resolver para utilizar VIP 10.10.10.1 para dominios bloqueados.
- Creación de subinterfaz enp0s8.10 (192.168.10.50/24) en Lubuntu para VLAN IoT.
- Creación de reglas de firewall para permitir DNS y acceso a Internet sólo desde VLAN10_IOT.
- Configuración de Outbound NAT en modo híbrido para 192.168.10.0/24.
- Verificación de bloqueos mediante dig y ping desde cliente IoT.

Se monitorizó la actividad de pfBlockerNG, verificando que las IPs maliciosas y dominios no deseados eran bloqueados adecuadamente, asegurando la protección activa de la red doméstica.

(El resto del documento continúa igual desde la sección 6.4 hasta el final, incluyendo en Anexos nuevas capturas de la configuración de OpenVPN, logs de conexión y dashboard de pfBlockerNG, además del dig y bloqueo de IP)

6.4. Fase 4: Instalación y configuración de Home Assistant e integración MQTT

En esta fase se instala Home Assistant en la red VLAN10 (IoT) y se configura su integración con el protocolo MQTT mediante Mosquitto, permitiendo gestionar sensores y automatizaciones.

6.4.1. Creación de máquina virtual

Se descargó e integró la imagen oficial de Home Assistant en formato.vdi. La configuración de la máquina virtual de Home Assistant se explicó previamente en el punto 5.3.

Al arrancar, recibió una IP por DHCP desde pfSense. Posteriormente se asignó una reserva estática con los parámetros anteriormente mencionados en el punto 6.1.2.

Esta configuración garantiza que Home Assistant tenga siempre la misma dirección IP dentro de la red de administración, y se ha decidido ubicar en la VLAN1 porque actúa como núcleo del sistema domótico y requiere acceso controlado, seguro y centralizado. De esta forma, se protege su interfaz de configuración y su integración con servicios críticos como VPN, backup o IDS, separándolo del tráfico de los dispositivos IoT.

6.4.2 Configuración del broker Mosquitto en Ubuntu

Se instaló el broker Mosquitto en una máquina Ubuntu conectada a la VLAN10, según lo descrito en el punto 4.6. con el comando “sudo apt update && sudo apt install -y mosquitto mosquitto-clients”, esto instala tanto el servidor (broker) como los comandos mosquitto_pub y mosquitto_sub para simular sensores o probar conexiones.

Alternativamente, Home Assistant también permite instalar el broker Mosquitto como complemento directamente desde su panel. Para ello, se accede a Configuración > Complementos > Mosquitto Broker, se instala el complemento y se arranca el servicio. En este caso, Home Assistant se conecta al broker mediante el host “localhost”, simplificando la configuración inicial para entornos de prueba o demostración.

6.4.3. Vinculación de MQTT con Home Assistant

Para permitir la comunicación, se creó un usuario con acceso MQTT en Home Assistant (Configuración > Personas > Añadir usuario):

- Nombre: Dispositivos IoT

- Nombre de usuario: iot_user
- Contraseña: CHANGEME_STRONG_PASSWORD
- Grupo: Usuario (NO administrador)

Luego, se añadió la integración MQTT desde el panel (Configuración > Dispositivos y servicios > Añadir > MQTT).

Home Assistant detectó automáticamente al broker Mosquitto y quedó vinculado al sistema de automatización.

6.4.4. Simulación de sensores MQTT

Desde Ubuntu, se simularon sensores usando mosquitto_pub. Para registrar un sensor de temperatura visible en Home Assistant, ver simulación completa del sensor MQTT en el Anexo 9.3.1.

La entidad “Temperatura Casa” apareció automáticamente en el panel (Configuración > Dispositivos y servicios > Entidades), tras su publicación vía MQTT. Luego, se envió un valor desde terminal de Ubuntu para simular una lectura. El comando utilizado puede consultarse en el Anexo 9.3.2.

La entidad fue detectada y mostrada en el panel principal de Home Assistant (resumen) con el valor correspondiente.

También pueden integrarse dispositivos actuadores como luces. Por ejemplo, se puede definir una bombilla MQTT manualmente en el archivo “configuration.yaml”. La configuración YAML completa puede consultarse en el Anexo 9.3.3.

Para el control de esta bombilla virtual se creó un script que ejecuta el servicio “light.turn_on”, descrito en el Anexo 9.3.4.

Además de la bombilla virtual, se integró una bombilla física basada en ESP8266 con firmware Tasmota, comunicándose vía MQTT con el broker Mosquitto previamente

configurado. Esta integración se realizó sin depender de servicios en la nube, y su configuración técnica completa puede consultarse en el Anexo 9.3.3.

6.4.5. Integración de bombilla física Tasmota

Para validar el funcionamiento con dispositivos físicos reales, se integró una bombilla inteligente basada en el chip ESP8266 con firmware Tasmota. Esta bombilla fue configurada para conectarse directamente al broker MQTT local (Mosquitto), permitiendo su control desde Home Assistant sin depender de servicios en la nube. Esta integración no interfiere con la configuración anterior y complementa la validación del entorno domótico con un dispositivo físico real.

1. Conexión a la red Wi-Fi y acceso inicial

Al encender la bombilla, esta entró automáticamente en modo de emparejamiento. Desde un dispositivo móvil se accedió a la red Wi-Fi generada por el propio dispositivo (tasmota_XXXX) y se accedió a su interfaz web desde la dirección http://192.168.4.1. Allí se introdujeron los datos de conexión a la red Wi-Fi de la VLAN10 (IoT) previamente configurada en pfSense.

2. Asignación de IP y acceso al panel de configuración

Una vez conectada a la red de la VLAN10, la bombilla recibió una dirección IP dinámica desde el servidor DHCP de pfSense (http://192.168.10.105). Esta dirección fue localizada mediante la interfaz web del firewall o con herramientas como la app Fing. A continuación, se accedió al panel web de la bombilla para configurar el tipo de módulo ("Generic" o "Template") desde el menú Configuration > Configure Module, asegurando compatibilidad con el control vía MQTT.

3. Configuración de MQTT

Desde el panel web de Tasmota, se accedió a Configuration > Configure MQTT y se introdujeron los siguientes parámetros para enlazar la bombilla con el broker Mosquitto:

- Host: 192.168.1.10 (IP de Home Assistant)
- Port: 1883
- Client ID: tasmota_bombilla_vlan10
- Topic: casa/luz/salon
- User: iot_user
- Password: CHANGEME_STRONG_PASSWORD

Además, se activó la opción de retención (SetOption0 1) desde la consola para asegurar la persistencia del último estado.

4. Integración manual en Home Assistant

En el archivo configuration.yaml de Home Assistant, se añadió la configuración correspondiente al dispositivo MQTT siguiendo el esquema de luces. El script completo puede consultarse en el Anexo 9.3.3.
Tras guardar los cambios, se validó la configuración desde Configuración > Controles del servidor y se reinició Home Assistant. La entidad “light.luz_salon” apareció correctamente en el panel de dispositivos, reflejando el estado y permitiendo el control remoto de la bombilla Tasmota desde la interfaz

5. Control desde la interfaz web

Se añadió una tarjeta de tipo “Luz” en el dashboard principal de Home Assistant para controlar el encendido y apagado de la bombilla desde la interfaz gráfica. La respuesta fue inmediata, confirmando la integración completa con el entorno MQTT y el control bidireccional desde la red local.
Esta prueba demuestra que el sistema domótico es capaz de gestionar dispositivos físicos reales sin depender de plataformas externas, asegurando privacidad, autonomía y control completo sobre la red doméstica inteligente. Además, permite que la bombilla física se utilice en automatizaciones o scripts en conjunto con dispositivos simulados o sensores virtuales, integrándose sin interferir en la configuración ya establecida del sistema.

6.4.6. Creación de automatizaciones reactivas

Se configuró una automatización para mostrar una alerta persistente si la temperatura supera los 28 °C, utilizando como disparador la entidad “sensor.temperatura_casa”. La configuración completa y el comando usado para su validación pueden consultarse en el Anexo 9.3.5.

También se programó una automatización diaria para encender la bombilla a las 18:10, como se detalla en el Anexo 9.3.6.

6.4.7. Automatización de copias de seguridad en Home Assistant

Además de sensores, actuadores y automatizaciones, se configuró una copia de seguridad automatizada del sistema para garantizar la integridad del entorno domótico desde la interfaz de Home Assistant. Esta automatización permite generar copias completas del sistema sin intervención manual. La automatización completa puede consultarse en el Anexo 9.3.7.

6.5. Fase 5: Simulación en Cisco Packet Tracer

6.5.1. Dispositivos utilizados y justificación

Se diseñó una topología de red lógica en Packet Tracer, con los siguientes dispositivos:

- Router Cisco ISR 2911 – Para configurar subinterfaces y actuar como gateway.
- Switch Cisco 2960 – Para segmentar VLANs y realizar trunking.
- 2 Access Points (IoT y invitados) – Para emitir redes Wi-Fi separadas.
- Dispositivos finales (Endpoints):
    - Garage Door (IoT) conectado a VLAN10
    - Smartphone (invitado) conectado a VLAN20

6.5.2. Topología de red

6.5.3. Configuración del router ISR

Se crearon subinterfaces G0/1.30, G0/1.10 y G0/1.20 para VLAN1, VLAN10 y VLAN20 respectivamente. Y se asignaron las siguientes direcciones IP:

- VLAN1: 192.168.1.1
- VLAN10: 192.168.10.1
- VLAN20: 192.168.20.1

Se crearon grupos DHCP y reglas NAT para el acceso a Internet.

6.5.4. Configuración del switch y VLANs

Se crearon las VLANs 1 (Administración), 10 (IoT) y 20 (invitados).

El puerto conectado al router se configuró como trunk.

Los puertos de los Access Points se configuraron como access en su VLAN correspondiente.

7. Resultados obtenidos

Una vez finalizada la instalación, configuración y simulación de la red segmentada con pfSense y Home Assistant, se realizaron varias pruebas funcionales para determinar si se habían alcanzado los objetivos. Los resultados confirman que la red funciona, es segura y que cada uno de sus segmentos actúa de manera coherente

7.1. Aislamiento efectivo de redes

Una de las pruebas principales consistió en comprobar la segmentación real entre la VLAN1 (Administración), la VLAN10 (IoT) y la VLAN20 (invitados) de la siguiente manera:

- Los dispositivos de la VLAN10 no pudieron acceder a la VLAN1 ni a la VLAN20, ni viceversa, mediante ping.
- pfSense bloqueó correctamente todo el tráfico entre las VLANs en el nivel de capa 3, tal y como se configuró en las reglas.
- Cada segmento era independiente del otro para el acceso a Internet.

Este comportamiento muestra que la segmentación fue exitosa, cumpliendo el principio de aislamiento de tráfico y reduciendo el riesgo de ataques laterales.

7.2. Pruebas de conectividad

Se verificó la conectividad de todos los dispositivos a su respectiva puerta de enlace y la salida a Internet para cada VLAN.

El cliente Ubuntu en VLAN10 obtuvo su IP dinámicamente (192.168.10.101) y se envió el siguiente ping:

- 192.168.10.1 (gateway)
- 192.168.1.10 (Home Assistant)
- Direcciones externas como 8.8.8.8 (verification de salida a Internet)

Para esta prueba, se configuró la subinterfaz VLAN de la siguiente manera:

sudo ip link add link enp0s3 name enp0s3.10 type vlan id 10

sudo ip link set enp0s3 up

sudo ip link set enp0s3.10 up

sudo dhclient enp0s3.10

Una vez asignada la IP por DHCP, se probó también el acceso HTTP a Home Assistant con el siguiente comando:

curl http://192.168.1.10:8123

El acceso fue exitoso, demostrando que las reglas de firewall funcionaban correctamente: el cliente solo podía comunicarse con el gateway y con el servicio específico de Home Assistant.

Al mismo tiempo, otro dispositivo simulado en VLAN20 recibió su IP 192.168.20.101 y también accedió a internet sin problemas.

Estas pruebas demostraron:

- El correcto funcionamiento de los servidores DHCP en cada VLAN.
- El aislamiento entre segmentos de red.
- El filtrado adecuado del tráfico por parte de pfSense.
- El acceso selectivo desde la VLAN IoT al servicio de automatización.

7.3. Acceso web funcional a Home Assistant

Desde una máquina Ubuntu conectada a la VLAN10 (IoT), se realizaron varias pruebas de conectividad para validar el acceso a Home Assistant:

- Se realizó un ping exitoso a la dirección IP 192.168.1.10, confirmando la ruta de red desde VLAN10 hacia la IP del servicio.
- Se accedió vía navegador web a la URL http://192.168.1.10:8123
- Se completó la configuración inicial mediante el asistente visual de Home Assistant, incluyendo la creación del usuario administrador.

- El panel de control se cargó correctamente y se añadieron varias integraciones, además de definir automatizaciones básicas.

La carga fue fluida y la interfaz funcionó sin problemas, lo que indica que Home Assistant fue instalado correctamente y era accesible dentro del entorno virtual segmentado.

7.4. Evaluación de seguridad y funcionamiento

Se evaluaron múltiples aspectos de seguridad y estabilidad del entorno: Separación efectiva de dominios de broadcast mediante VLAN.

- Accesos restringidos entre redes mediante reglas de firewall personalizadas en pfSense.
- Asignación controlada de direcciones IP mediante servidores DHCP independientes.
- Acceso al panel de pfSense limitado exclusivamente a la red de administración (VLAN1).
- Implementación de pfBlockerNG para bloqueo de dominios e IPs maliciosos, reforzando la seguridad perimetral.

Además, se instaló OpenVPN para permitir el acceso remoto cifrado a la red, lo que añade una capa adicional de seguridad para administración remota.

La respuesta del entorno virtual fue estable, sin errores de conexión ni caídas durante las pruebas.

La simulación en Packet Tracer reforzó la validación de la topología lógica, confirmando que la configuración aplicada en el entorno virtual se ajustaba al diseño planificado.

7.5. Prueba de integración de dispositivo físico Tasmota

Como parte de la validación funcional, se conectó una bombilla física con firmware Tasmota al entorno domótico. La bombilla fue enlazada mediante MQTT al broker Mosquitto y añadida manualmente en Home Assistant. Su encendido, apagado y

automatizaciones funcionaron correctamente desde la interfaz gráfica, lo que confirma la integración real de dispositivos físicos en la arquitectura propuesta. Los detalles de la configuración están disponibles en el Anexo 9.3.3.

8. Conclusiones

El desarrollo del proyecto permitió verificar la factibilidad del diseño e implementación de una red doméstica inteligente y segmentada basada en herramientas de código abierto como pfSense y Home Assistant utilizando buenas prácticas en seguridad y administración de redes.
La experiencia adquirida fue muy enriquecedora desde ambos extremos: técnico, de planificación y análisis de riesgo, y de documentación de entornos de TI.

8.1. Logros alcanzados

Se ha configurado una infraestructura de red segmentada mediante VLANs, asegurando la separación lógica entre la rede de administración, dispositivos IoT y usuarios invitados.

Se ha instalado y puesto en funcionamiento pfSense como firewall avanzado, gestionando el tráfico de red, asignación de IPs y reglas de seguridad entre VLANs.

Home Assistant se ha desplegado con éxito en la red de administración, validando su integración con dispositivos de la VLAN IoT y demostrando su viabilidad como núcleo de un sistema domótico centralizado.

Asimismo, se ha logrado una simulación realista en Cisco Packet Tracer, reforzando el entendimiento de la topología física y lógica del entorno propuesto.

El entorno desarrollado es funcional, seguro, replicable y fácilmente ampliable, tanto en contextos educativos como reales.

La integración del broker MQTT Mosquitto con Home Assistant ha permitido implementar una plataforma de automatización escalable y controlable. Gracias a sensores simulados

mediante comandos y su visualización en el panel principal (resumen), se han creado automatizaciones reactivas, notificaciones inteligentes y acciones coordinadas en función de eventos del entorno detectados en la red domótica.

Se logró integrar y controlar tanto una bombilla virtual como una física con firmware Tasmota, ambas gestionadas mediante el protocolo MQTT desde Home Assistant. Esta integración valida la capacidad del sistema para interactuar de forma real con actuadores físicos sin depender de plataformas externas, fortaleciendo la funcionalidad y autonomía del entorno domótico. La configuración completa se recoge en el Anexo 9.3.3.

Finalmente, se configuró una automatización para realizar copias de seguridad diarias del sistema, reforzando su estabilidad y asegurando la capacidad de recuperación ante errores o fallos inesperados.

8.2. Limitaciones encontradas

Aunque se integró una bombilla física con firmware Tasmota, el uso de dispositivos físicos de IoT en general ha sido limitado debido a las restricciones del entorno virtualizado. Esto implica que no se pudieron comprobar todas las interacciones posibles con sensores o actuadores reales. La simulación en Packet Tracer permitió representar la lógica de red, pero no sustituye completamente el comportamiento dinámico de dispositivos físicos.

Se implementó acceso remoto seguro mediante OpenVPN, así como mecanismos de protección adicionales como pfBlockerNG, que permitió bloquear dominios e IPs maliciosos. Sin embargo, no se añadieron aún medidas avanzadas como certificados mutuos de cliente, autenticación multifactor o herramientas de monitorización en tiempo real como IDS/IPS, que sí serían recomendables en un entorno productivo.

Por último, Home Assistant presenta una curva de aprendizaje inicial al configurar automatizaciones avanzadas, aunque su versatilidad lo convierte en una plataforma muy adecuada para entornos educativos y de integración real.

8.3. Propuestas de mejora

Para futuras versiones o ampliaciones de este proyecto, se proponen las siguientes líneas de mejora:

- Ampliar la implementación de dispositivos físicos IoT, integrando sensores de movimiento, enchufes inteligentes, cerraduras electrónicas o cámaras IP para validar una gama más amplia de automatizaciones.
- Mejorar la configuración de Home Assistant con automatismos más complejos, paneles de control personalizados para diferentes perfiles de usuario y reglas condicionales avanzadas.
- Reforzar la monitorización de seguridad en tiempo real, incorporando herramientas como ntopng, Snort o Suricata (IDS/IPS) para analizar patrones de tráfico y detectar intrusiones.
- Ampliar la autenticación de acceso remoto, integrando autenticación multifactor (MFA) en OpenVPN o certificados digitales personalizados.
- Documentar copias de seguridad cruzadas y automatizadas, tanto en Home Assistant como en pfSense, asegurando la continuidad ante fallos o pérdida de datos.

Aunque pfBlockerNG y OpenVPN ya han sido integrados en esta versión, podrían profundizarse sus configuraciones para mayor robustez en un entorno de producción.

8.4. Conclusión personal

En nuestra opinión personal, este proyecto es valioso tanto a nivel técnico como académico. Primero, tuvimos la oportunidad de comprobar los conocimientos adquiridos sobre el ciclo ASIR en un entorno real. Nos parecía interesante realizar la planificación, diseño de redes en la nube así como instalar y configurar sistemas de automatización doméstica. En particular, el uso de herramientas pfSense y Home Assistant nos permitió investigar con mayor detalle sobre los principales conceptos, incluyendo la segmentación de redes, la seguridad perimetral y la orquestación de dispositivos IoT. A nuestro modo

de ver, el proyecto es un excelente punto de contacto entre la teoría y la práctica, preparándonos para los futuros desafíos técnicos y profesionales relacionados con los campos de administración de sistemas y ciberseguridad en un sentido amplio.

9. Anexos

Esta sección recopila documentación adicional relevante para comprender la implementación técnica del proyecto. Se incluyen capturas de pantalla de las configuraciones más críticas, transcripciones de comandos y planos generados por Packet Tracer; esto tiene como objetivo dar transparencia, trazabilidad y profundidad técnica al trabajo.

9.1. Capturas de instalación

Se incluyen capturas de pantalla de las fases de instalación y configuración del entorno virtualizado:

Figura 1

Pantalla de arranque de pfSense en VirtualBox con asignación de interfaces (em0, em1, em1.10, em1.20).

Figura 2

Pantalla de inicio de sesión en la interfaz web de pfSense desde VirtualBox (192.168.1.1).

Figura 3

Asignación de subinterfaces em1.10 (VLAN10_IoT) y em1.20 (VLAN20_INV) en em1 (LAN).

Figura 4

Asignación de interfaz em0 (WAN), em1 (LAN), em1.10 (VLAN10_IoT) y em1.20 (VLAN20_INV).

Figura 5

Configuración del servidor DHCP en pfSense para la red de administración (LAN).

Figura 6

Configuración del servidor DHCP para la red de dispositivos IoT (VLAN10_IOT).

Figura 7

Configuración del servidor DHCP para la red de invitados (VLAN20_INV).

Figura 8

Reglas predeterminadas del firewall en la interfaz LAN.

Figura 9

Reglas del firewall aplicadas en la VLAN10_IOT para bloqueo entre redes y acceso a Internet.

Figura 10

Reglas del firewall aplicadas en la VLAN20_INV para aislamiento de red y acceso a Internet.

Figura 11

Reglas del firewall aplicadas en OpenVPN para aislamiento de red y acceso a Internet.

Figura 12.

Creación de la CA interna en pfSense para OpenVPN.

Figura 13.

Configuración del servidor OpenVPN en pfSense.

Figura 14.

Exportación del perfil cliente OpenVPN.

Figura 15.

Conexión OpenVPN en cliente Linux (Initialization Sequence Completed).

Figura 16.

Resultados mostrando interfaz tun0.

Figura 17.

Prueba de conectividad con ping a IP LAN de pfSense.

Figura 18.

Pantalla inicial de pfBlockerNG en pfSense.

Figura 19.

Configuración de IP Feeds en pfBlockerNG.

Figura 20.

Configuración de DNSBL en pfBlockerNG.

Figura 21.

Resultado del comando dig doubleclick.net @192.168.10.1

Figura 22.

Estadísticas de bloqueos en pfBlockerNG: (a) IP Block Stats, (b) DNSBL Block Stats. Fuente: elaboración propia.

Nota: no se muestran logs de pfBlockerNG porque no se detectó tráfico bloqueado durante la fase de pruebas.

Figura 23.

Log de tráfico bloqueado por pfBlockerNG en pfSense.

Figura 24.

Realización de copias de seguridad en pfSense.

Figura 25.

Pantalla de arranque de Home Assistant en VirtualBox con dirección IP asignada (192.168.1.10).

Figura 26.

Asignación de IP estática de Home Assistant desde pfSense (reserva DHCP: 192.168.1.10)

Figura 27.

Pantalla de inicio de sesión en la interfaz web de Home Assistant desde virtualbox (192.168.1.10:8123).

Figura 28.

Pantalla de configuración web de la bombilla Tasmota (http://192.168.1.x)

Figura 29.

Interfaz MQTT en Home Assistant detectando la bombilla (panel de dispositivos)

Figura 30.

Panel de Home Assistant con la bombilla encendida

Figura 31.

Panel de Home Assistant con la bombilla apagada

mqtt:

light:

- name: "Luz Salon"

state_topic: "casa/luz/salon/state"

command_topic: "casa/luz/salon/set"

payload_on: "ON"

payload_off: "OFF"

qos: 0

retain: true

optimistic: false

Figura 32.

Script YAML o interfaz gráfica mostrando la automatización de la luz

Figura 33.

Panel de copias de seguridad mostrando una realizada a las 03:00

- Las figuras están numeradas como Figura 1, Figura 2, etc., y cada una tendrá su leyenda de acuerdo con APA 7.

9.2. Configuraciones completas

Las configuraciones clave realizadas en los diferentes entornos son:

pfSense

- Interfaces asignadas y configuradas.
- Servidores DHCP para VLAN1, VLAN10 y VLAN20.
- Reglas de firewall: bloqueo inter-VLAN, acceso a Internet, acceso al panel.
- Configuración de servicios adicionales (NAT, DNS Resolver, etc.).

Configuración de subinterfaces VLAN en pfSense:

Desde el panel web de pfSense (https://192.168.1.1), se añadieron interfaces VLAN virtuales sobre la interfaz troncal “em1”, desde Interfaces > Assignments > VLANs. Las interfaces configuradas fueron:

- VLAN10_IOT (ID 10): 192.168.10.0/24 - Dispositivos IoT
- VLAN20_INV (ID 20): 192.168.20.0/24 - Red de invitados

Cada VLAN fue asignada como interfaz virtual, con su dirección IP y un servidor DHCP independiente. Esta segmentación lógica facilitó la gestión de dispositivos por tipo de usuario y la aplicación de políticas de seguridad diferenciadas.

Configuración de reglas de firewall en pfSense:

Cada interfaz de red en pfSense (LAN, VLAN10_IOT y VLAN20_INV) fue configurada con reglas de firewall personalizadas desde el panel web de pfSense (https://192.168.1.1) en el menú Firewall > Rules. Estas reglas se aplicaron con el objetivo de segmentar el tráfico y reforzar la seguridad entre las distintas VLANs.

Para la red VLAN10_IOT:

- Se permitió acceso a su gateway (192.168.10.1)
- Se permitió acceso a Home Assistant ubicado en la red de administración (192.168.1.10)
- Se bloqueó acceso a red de administración (192.168.1.0/24)
- Se bloqueó acceso a red de invitados (192.168.20.0/24)

Configuración de servicios adicionales en pfSense:

Además de la asignación de IPs, VLANs y reglas de firewall, se activaron servicios clave:

• NAT (Network Address Translation):

Se configuró desde Firewall > NAT para permitir que las redes internas (192.168.1.0/24, 192.168.10.0/24 y 192.168.20.0/24) accedan a Internet mediante la interfaz WAN.

• DNS Resolver:

Activado por defecto. Permite a pfSense resolver nombres de dominio para todos los clientes de red. Se accedió desde Services > DNS Resolver, configurado en modo estándar.

• WebConfigurator (interfaz de administración):

Se habilitó el acceso a la administración de pfSense exclusivamente desde la red de administración (LAN), mediante HTTPS. También se configuró la redirección automática desde HTTP a HTTPS para mayor seguridad.

Para la red VLAN20_INV:

- Se permitió acceso a su gateway (192.168.20.1)
- Se permitió acceso a Internet
- Se bloqueó acceso a red de administración (192.168.1.0/24)

- Se bloqueó acceso a red de dispositivos IoT (192.168.10.0/24)

Estas reglas permiten aislar los distintos tipos de tráfico según el tipo de usuario o dispositivo, evitando accesos laterales entre segmentos, y garantizando el principio de mínimo privilegio en el diseño de red.

Home Assistant

- Configuración de red (IP fija por DHCP).
- Interfaz inicial y configuración básica del sistema.
- Creación de usuario administrador.
- Exploración del panel de integraciones.

Cisco Packet Tracer

- Script completo del router ISR 2911: subinterfaces, NAT, pools DHCP.
- Configuración del Switch 2960: VLANs, puertos access, trunk.
- Diagramas lógicos y conexión entre dispositivos.
- Todos los fragmentos de código y comandos estarán numerados como "Código 1", "Código 2", etc., con sus respectivos títulos.

9.3. Código y scripts de automatización en Home Assistant

Fragmentos de código y comandos utilizados en Home Assistant y dispositivos relacionados, tanto para configuración como para automatizaciones, que permitieron el control de elementos físicos y virtuales dentro del entorno domótico.

9.3.1. Simulación de sensor de temperatura (mosquitto_pub)

Para verificar la recepción de datos en Home Assistant desde un sensor MQTT simulado, se utilizó el comando “mosquitto_pub” desde la terminal de una máquina Ubuntu conectada a la VLAN IoT. Este comando permite publicar mensajes MQTT en un tópico

concreto, simulando así una lectura de temperatura como si proviniera de un sensor físico.

La primera instrucción configura el sensor en Home Assistant utilizando un payload en formato JSON. La segunda instrucción envía un valor real (en este caso, 24.6 °C) al tópico previamente definido, permitiendo observar el valor en el panel web del sistema domótico.

mosquitto_pub -h 192.168.1.10 -u iot_user -P CHANGEME_STRONG_PASSWORD -t

"homeassistant/sensor/temperatura_casa/config" -m '{

"name": "Temperatura Casa",

"state_topic": "casa/sensor/temperatura",

"unit_of_measurement": "°C",

"device_class": "temperature"

}'

9.3.2. Simulación de sensor de temperatura (mosquitto_pub)

Se publica un valor simulado (24.6 °C) en un topic MQTT desde la terminal de Ubuntu para que Home Assistant lo lea desde un sensor de temperatura virtual.

mosquitto_pub -h 192.168.1.10 -u iot_user -P CHANGEME_STRONG_PASSWORD -t "casa/sensor/temperatura" -m "24.6"

9.3.3. Configuración de bombilla virtual y física (configuration.yaml)

Para la simulación y automatización de luces se configuraron dos bombillas: una virtual y otra física con firmware Tasmota, ambas gestionadas mediante el protocolo MQTT.

La bombilla virtual se configuró con el objetivo de validar flujos de automatización desde Home Assistant sin necesidad de hardware físico. Esta entidad fue declarada manualmente en el archivo configuration.yaml, utilizando el esquema MQTT light.

La segunda bombilla es un dispositivo físico basado en el microcontrolador ESP8266 y con firmware Tasmota, configurada también mediante MQTT. Esta integración permite un control bidireccional local, sin dependencia de servicios en la nube

```
mqtt:
  light:
    - name: Luz Salon
      state_topic: casa/luz/salon/state
      command_topic: casa/luz/salon/set
      payload_on: ON
      payload_off: OFF
      qos: 0
      retain: true
      optimistic: false

    - name: Bombilla Tasmota
      state_topic: stat/bombilla_salon/POWER
      command_topic: cmnd/bombilla_salon/POWER
      payload_on: "ON"
```

```
    payload_off: "OFF"

    qos: 1

    retain: true
```

Esta configuración conjunta en el archivo configuration.yaml permite a Home Assistant detectar y controlar tanto dispositivos simulados como físicos reales, facilitando las pruebas y automatizaciones del entorno domótico implementado.

9.3.4. Script de encendido de bombilla (scripts.yaml)

Este script permite encender la bombilla virtual definida previamente en el archivo “configuration.yaml”, utilizando el servicio “light.turn_on”. Se ha declarado dentro del archivo “scripts.yaml”, donde se pueden crear acciones reutilizables que luego pueden ser ejecutadas manualmente o desde automatizaciones.

El alias “Encender Luz Salón” es el nombre que aparecerá en la interfaz de Home Assistant, permitiendo activarlo directamente desde el panel de forma manual o integrarlo en flujos automatizados.

```
encender_luz_salon:
  alias: Encender Luz Salón
  sequence:
    - service: light.turn_on
      target:
        entity_id: light.luz_salon
  mode: single
```

9.3.5. Automatización de alerta por temperatura (automations.yaml)

Desde la interfaz de Home Assistant se creó una automatización que muestra una notificación persistente cuando el sensor “sensor.temperatura_casa” detecta un valor superior a 28 °C. Esta acción se configura desde Configuración > Automatizaciones y escenas > Crear automatización.

```
alias: Alerta temperatura alta

trigger:

  - platform: numeric_state

    entity_id: sensor.temperatura_casa

    above: 28

    for: "01:00:00"

action:

  - service: persistent_notification.create

    data:

      title: Temperatura excesivamente alta

      message: "¡Alerta! La temperatura ha superado los 28 °C."

mode: single
```

Para comprobar el funcionamiento de esta automatización, se publicó un valor simulado (28.5 °C) en un topic MQTT desde la terminal de Ubuntu:

```
mosquitto_pub -h 192.168.1.10 -u iot_user -P CHANGEME_STRONG_PASSWORD -t "casa/sensor/temperatura" -m "28.5"
```

9.3.6. Automatización programada de encendido de bombilla (automations.yaml)

Esta automatización permite encender automáticamente la bombilla virtual “light.luz_salon” cada día a las 18:10.

```
alias: Encender bombilla 18:10

trigger:

  - platform: time

    at: "18:10:00"

action:

  - service: light.turn_on

    target:

      entity_id: light.luz_salon

mode: single
```

9.3.7. Automatización de copia de seguridad (automations.yaml)

Automatización en Home Assistant que genera una copia de seguridad diaria a las 03:00.

```
alias: Backup diario

trigger:

  - platform: time

    at: "03:00:00"

action:

  - service: backup.create
```

mode: single

9.4 Configuración de dispositivos en Cisco Packet Tracer

Configuración de red simulada en el entorno Cisco Packet Tracer, incluyendo router, switch, NAT y DHCP.

9.4.1. Configuración del Router ISR 2911

interface GigabitEthernet0/1.10

encapsulation dot1Q 10

ip address 192.168.10.1 255.255.255.0

ip nat inside

interface GigabitEthernet0/1.20

encapsulation dot1Q 20

ip address 192.168.20.1 255.255.255.0

ip nat inside

interface GigabitEthernet0/1.30

encapsulation dot1Q 30

ip address 192.168.1.1 255.255.255.0

ip nat inside

interface GigabitEthernet0/1

no shutdown

9.4.2. Pools DHCP por VLAN

ip dhcp pool VLAN10_IOT

```
 network 192.168.10.0 255.255.255.0
 default-router 192.168.10.1
 dns-server 8.8.8.8
ip dhcp pool VLAN20_INV
 network 192.168.20.0 255.255.255.0
 default-router 192.168.20.1
 dns-server 8.8.8.8
ip dhcp pool VLAN30_ADMIN
network 192.168.1.0 255.255.255.0
default-router 192.168.1.1
dns-server 8.8.8.8
```

9.4.3. Reglas NAT para acceso a Internet (G0/0)

```
access-list 1 permit 192.168.10.0 0.0.0.255
access-list 1 permit 192.168.20.0 0.0.0.255
access-list 1 permit 192.168.1.0 0.0.0.255
interface GigabitEthernet0/0
ip address dhcp
ip nat outside
ip nat inside source list 1 interface GigabitEthernet0/0 overload
```

9.4.4. Configuración del Switch 2960 (VLANs y puertos)

```
vlan 10
```

name IoT

vlan 20

name Inv

vlan 30

name Admin

Puerto Trunk hacia el router:

interface FastEthernet0/1

switchport mode trunk

Puertos Access por VLAN:

interface FastEthernet0/2

switchport mode access

switchport access vlan 10

interface FastEthernet0/3

switchport mode access

switchport access vlan 20

interface FastEthernet0/4

switchport mode access

switchport access vlan 30

Estos comandos están comentados y organizados en el anexo para su consulta detallada.

10. Referencias bibliográficas (formato APA 7)

En conclusión, se presentan las fuentes utilizadas en el desarrollo del presente trabajo, basadas en documentación oficial, bibliografía técnica, guías académicas y recursos de educación.

pfSense. (2024). Documentación y descargas de pfSense. Recuperado el 5 de abril de 2025, de https://www.pfsense.org/

Home Assistant. (2024). Getting started with Home Assistant. Recuperado el 5 de abril de 2025, de https://www.home-assistant.io/docs/

Cisco Networking Academy. (2024). Cisco Packet Tracer Labs and Tutorials. Cisco Systems. Recuperado de https://www.netacad.com/

Netgate. (2024). Installing pfSense on VirtualBox. Recuperado el 6 de abril de 2025, de https://docs.netgate.com/

[Institución]. (2024). Guía de citación y estilo APA 7ª edición. Recuperado de https://web-uem.bibliocrai.universidadeuropea.com/200-escuelas-y-facultades/todas/725-estilo-apa#video-apa-7%C2%AA-ed

Hernández, J., & Pérez, M. (2021). Redes de computadoras y seguridad en entornos IoT. Editorial RA-MA.

Badillo, J. (2017). Sistemas de automatización del hogar: tecnologías aplicadas. Ediciones Alfaomega.

Ischimji, K., Ramírez, A., & Ortega, D. (2012). Guía práctica de diseño de redes LAN segmentadas. Universidad Politécnica de Madrid.

Eclipse Mosquitto. (2024). Mosquitto MQTT Broker Documentation. Recuperado de https://mosquitto.org/documentation/

Tasmota Project. (2024). Tasmota - Documentation. Recuperado de https://tasmota.github.io/docs/

OpenVPN Technologies. (2024). OpenVPN Installation Guide. Recuperado de https://openvpn.net/community-resources/how-to/

OpenAI. (2024). ChatGPT (versión GPT-4), modelo de lenguaje para generación de scripts y asistencia técnica en redacción técnica y configuración de dispositivos. Recuperado de https://chat.openai.com/