



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

CAMPUS D'ALCOI

CPE WAN Management Protocol (CWMP)

Memoria Presentada Por:

James Henry Mervyn Hopkins

Director:

Raúl Llinares Llopis

Grado de Ingeniería Informática

Convocatoria de Defensa: Junio 2018

En el Proyecto se ha desarrollado un sistema Auto-Configuration Server (**ACS**), un sistema capaz de monitorizar, gestionar y aprovisionar configuraciones y actualizaciones de software y firmware preestablecidas a dispositivos cliente (**CPE**) remotos. Un sistema ACS realiza uso de CPE WAN Management Protocol (**CWMP**), un protocolo para el aprovisionamiento de configuraciones a CPEs.

Diseñamos una red de pruebas para simular cómo funcionaría nuestro servicio ACS (**GenieACS**) en una empresa proveedor de servicios de Internet (**ISP**). Para nuestras pruebas, la red se diseñó con las tecnologías FTTH y Wireless/WIMAX para observar el uso del sistema ante ambos casos.

Configuramos unos estados (**Preset**) que contienen configuraciones que se pueden aprovisionar a un CPE dependiendo de unas condiciones previas preestablecidas.

Finalmente, conectamos un CPE a nuestra red de prueba y observamos que recibía la configuración de todos los Presets, acabando con una configuración permitiendo al CPE acceso al Internet.

In this Project I have developed an Auto-Configuration Server (**ACS**), a system capable of monitoring, administering and provisioning configurations and pre-established updates of software and firmware to remote client devices (**CPE**). An ACS system uses the CPE WAN Management Protocol (**CWMP**), a protocol for provisioning configurations to CPEs.

I designed a test network to simulate how the ACS system (**GenieACS**) would work in an internet service provider (**ISP**). For the tests, the network was designed with FTTH and Wireless/WIMAX technologies to observe how the system handled in both cases.

I then configured a group of states (**Preset**) that contain configurations that can be provisioned to a CPE according to pre-established preconditions.

Finally, I connected a CPE to the test network and observed that it received all the configurations from the Presets, allowing the CPE access to the Internet.

Palabras clave: CWMP; TR-069; GenieACS; WIMAX; FTTH

Tabla de Contenido

1. Introducción.....	4
1.1 Marco de Referencia.....	4
1.2 Justificación.....	4
1.3 Objetivo Principal.....	5
1.4 Estructura de la Memoria	5
2. Fundamentos Teóricos	6
2.1. Redes ISP.....	6
2.1.1. Tecnologías de Última Milla.....	11
2.1.2. AAA (Authentication, Authorization, and Accounting).....	19
2.2. Auto-aprovisionamiento.....	25
2.2.1. CWMP	26
2.2.2. ACS.....	27
2.2.3. SOAP.....	28
2.2.4. RPC.....	29
2.2.5. TR-098.....	31
3. Implementación Práctica.....	32
3.1. Topología de Red	32
3.1.1. Infraestructura de Red.....	33
3.1.2. Direccionamiento IP.....	35
3.2. Auto-aprovisionamiento.....	36
3.2.1. Servidor GenieACS	36
3.2.2. Presets Configurados	47
3.2.3. TFTP.....	53
3.2.4. Página Web	54
3.3. Aprovisionamiento de CPEs	55
3.3.1. Redes Ethernet/Wimax.....	56
3.3.2. Redes FTTH	60
3.4. Gestfy.....	64
4. Conclusiones y Futuras líneas de Trabajo	66
5. Bibliografía.....	68

1. Introducción

1.1 Marco de Referencia

Nuestro proyecto se sitúa en un entorno empresarial, concretamente en una empresa de **Telecomunicaciones** dedicada a la **provisión de servicios de Internet (ISP)** con unos 2000 clientes aproximadamente, donde mantienen un sistema para guardar los datos de dichos clientes, incluyendo los parámetros de conexión a Internet (**Accounting**).

En este contexto, el alumno de Ingeniería Informática, especialidad en Redes, **James Henry Mervyn Hopkins**, durante unas prácticas en empresa, realiza el estudio de un sistema **CPE WAN Management Protocol (CWMP)** y lleva a cabo la implementación de un servicio **Auto-Configuration Server (ACS)**. CWMP es un protocolo para aprovisionar configuraciones a dispositivos cliente (CPE), y ACS es un servicio capaz de gestionar dispositivos remotos para aprovisionar configuraciones, actualizaciones de software y firmware y monitorizar el estado de dichos dispositivos.

Todo esto con objetivo de dar soporte a la empresa para mantener mejor las configuraciones de los dispositivos finales de los clientes y auto-aprovisionarlos en caso necesario.

1.2 Justificación

En muchas empresas ISP, surge el problema de que los dispositivos finales de los clientes se **quedan sin la configuración inicial** que les ha proporcionado el operador, tanto por fallos del sistema como por error del propio cliente. Esto hace que el técnico tenga que intervenir, en ocasiones con desplazamiento al domicilio del cliente, para volver a configurar el dispositivo. Este tiempo, que repercute en la eficacia de los recursos humanos del operador, en ocasiones puede evitarse.

Muchos ISPs para evitar este problema deciden no dejar a sus clientes realizar modificaciones que podrían afectar negativamente el funcionamiento de los dispositivos, cerrando el acceso a ellos, aportando a los clientes **usuarios con menor privilegio**, con el fin de poder modificar algunos parámetros específicos muy básicos. Pero en muchos casos, los ISPs no quieren invertir tiempo pre-configurando un router de manera que el cliente no pueda realizar dichas modificaciones, quieren poder llevar un router totalmente nuevo y poder instalarlo sin ningún problema.

Para tener dicho servicio de poder llevar un router sin configuración inicial al cliente, conectarlo y que el servidor le mande la configuración apropiada al cliente, decidimos desarrollar este proyecto de un servicio ACS.

1.3 Objetivo Principal

El objetivo principal de este proyecto es aportar a los proveedores de servicios de Internet un servicio que pueda **monitorizar** los **Customer-Premises Equipment (CPE)** de la empresa y **aprovisionarles** archivos de configuración remotamente de manera automática.

En segundo lugar, se documentará la creación de **archivos de configuración generales** para todos los dispositivos, independiente del fabricante, y después **archivos de configuración específicos** para cada cliente con sus parámetros de conexión.

De esta manera el técnico podrá llevar un dispositivo al cliente sin configuración inicial, conectarlo a la red utilizando **cualquier tecnología** como ADSL, FTTH, WIMAX, etc. y que el servidor ACS aporte al dispositivo la configuración correcta.

Además, en caso de que hubiera algún fallo en el dispositivo, provocando que se quede sin la configuración inicial, el servicio implementado aportará una configuración preestablecida a dicho dispositivo.

1.4 Estructura de la Memoria

La memoria consistirá en 5 puntos diferenciados:

El **primer punto** explicará el porqué de este proyecto así como sus objetivos.

El **segundo punto** desarrollará los fundamentos técnicos necesarios para la realización de dicho proyecto.

El **tercer punto** consistirá en la implementación de nuestro servicio CWMP documentando toda la configuración que hemos realizado así como los criterios sobre los cuales nos hemos basado para dicha configuración.

El **cuarto punto** expondrá las conclusiones obtenidas durante el periodo de prácticas, aplicando lo previamente aprendido en la Escuela Politécnica Superior de Alcoy.

Y para terminar, el **quinto punto** citará las fuentes de documentación que se han utilizado, o al menos, la más relevante para la realización de este proyecto.

2. Fundamentos Teóricos

2.1. Redes ISP

Un proveedor de servicios de Internet, (ISP - **Internet Service Provider**), es la empresa que proporciona a los usuarios el **acceso a Internet** y los servicios relacionados. Un ISP conecta a sus usuarios a Internet a través de diferentes tecnologías, las tecnologías en los que nos centramos en este proyecto serán Wireless/Wimax y FTTH (GPON).

Los ISPs, normalmente, diseñan su red basándose en un **modelo de tres capas**: Capa de **Core**, Capa de **Distribución** y Capa de **Acceso**.

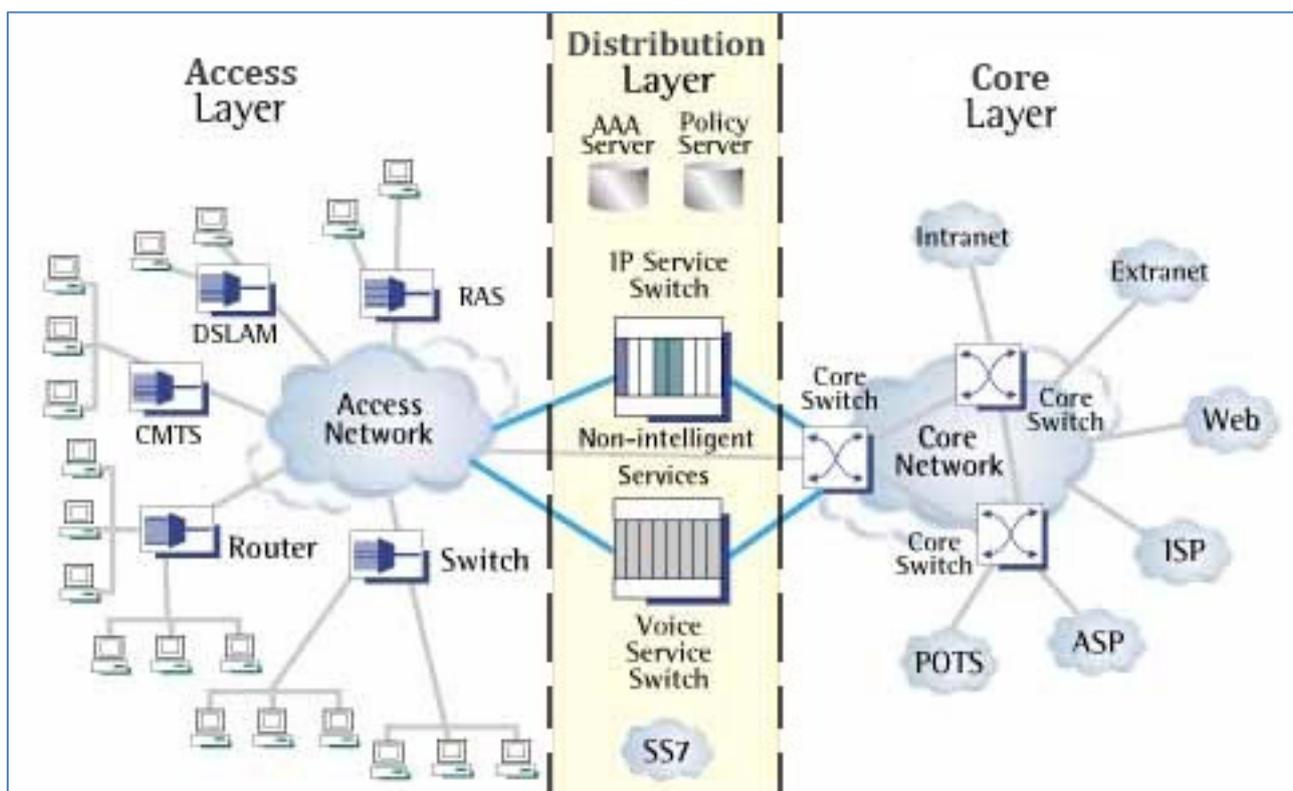


Figura 1 - Ejemplo Red ISP 3 Capas

La **Capa de Core** (Red Backbone) se considera la **red troncal** de la red, donde no se realiza filtrado de paquetes ni listas de acceso (**ACL**), con el objetivo de garantizar la entrega de paquetes de manera **rápida**. En esta capa tiene que haber **redundancia**, utilizado en caso de falla de enlace o hardware.

La **Capa de Distribución** (Red ISP) se encarga de garantizar que los paquetes se enruten correctamente entre las subredes y las VLAN (Redes Virtuales de Área Local) del ISP. En esta capa se suele implementar: **QoS** (Calidad de Servicio), **filtrados de paquetes**, **reglas de firewall**, etc.

La **Capa de Acceso** se enfoca en conectar dispositivos de cliente a la red ISP. También se utiliza para **filtrar direcciones MAC** e implementar equilibrio de carga (**Load Balancing**).

Un ejemplo de esquema de una red ISP, enfocada a la **capa de acceso**, sería lo siguiente:

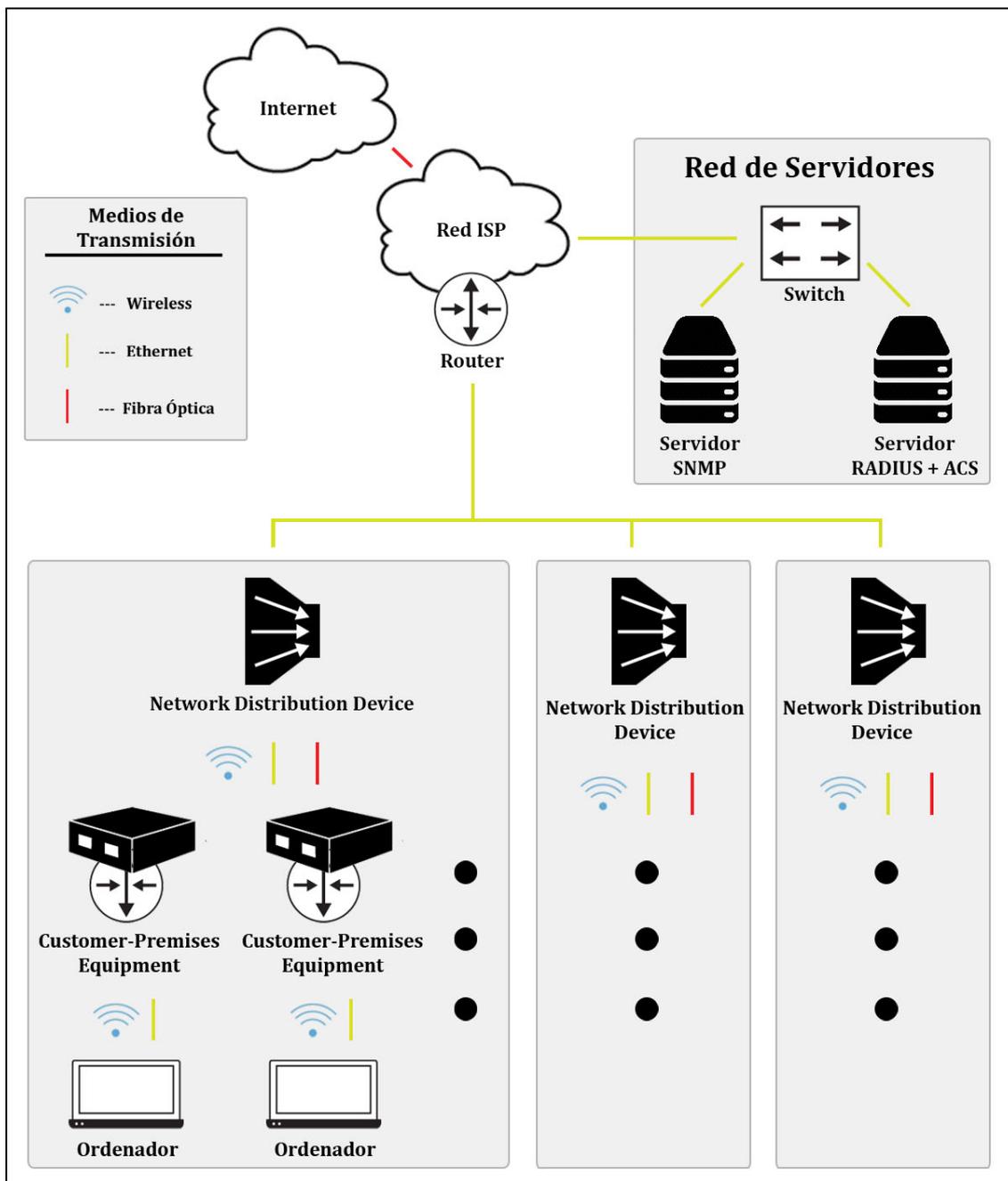


Figura 2 – Esquema típico de una Red ISP

Observamos la **Red de Servidores**, una zona de la red ISP donde se localizan los servidores. Estos servidores proporcionan los **servicios ofrecidos** por el ISP.

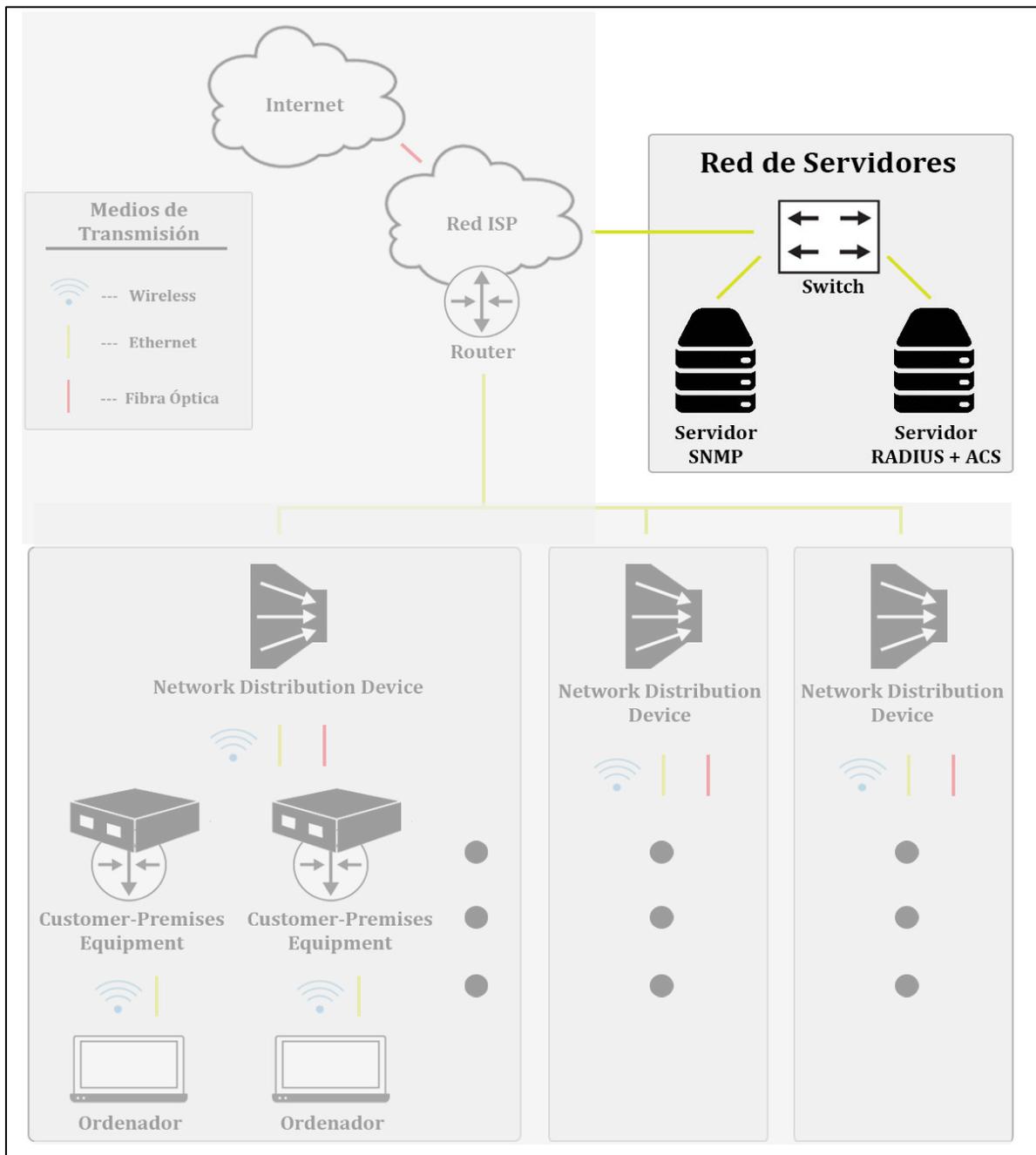


Figura 3 – Red de Servidores

En nuestro caso es donde se localiza un servidor Simple Network Management Protocol (**SNMP**) para monitorizar nuestros dispositivos (Switches y Routers) y nuestro servidor Remote Authentication Dial-In User Service + Auto-Configuration Server (**RADIUS + ACS**) para gestionar los usuarios y su uso de los servicios ofrecidos por el ISP.

Independiente de la tecnología de red que se utiliza en el ISP, encontraremos siempre dos dispositivos presentes: **Network Distribution Device** y **Customer-Premises Equipment (CPE)**.

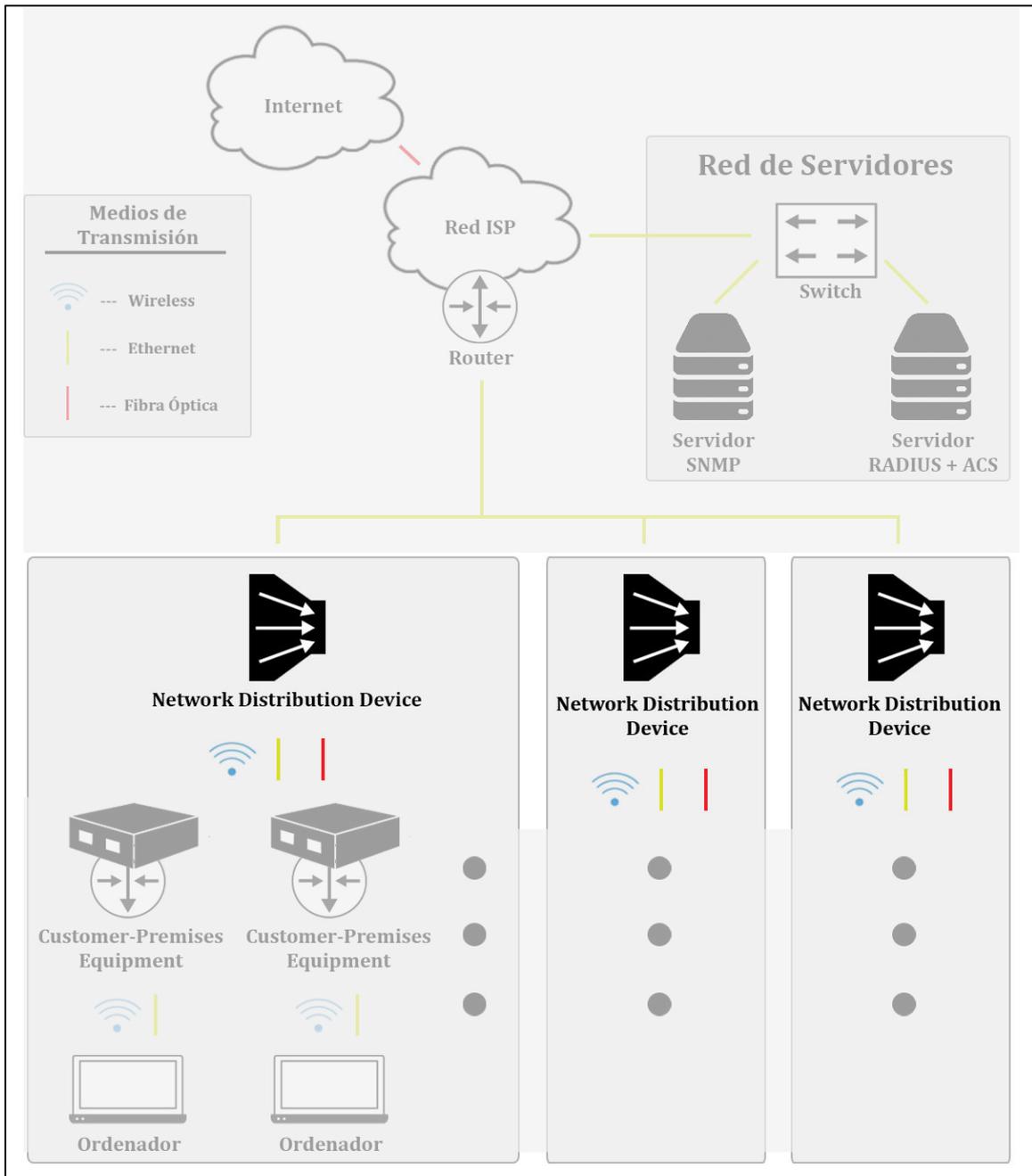


Figura 4 – Network Distribution Device

Un **Dispositivo de distribución de red (Network Distribution Device)** es capaz de agregar líneas de cliente individuales y transferir datos de todos sus clientes a un enlace uplink de alta velocidad que se conecta al resto de la **red de distribución** de un ISP mediante cualquier medio de transmisión. Este dispositivo también se conoce como **Network Access Control (NAC)**.

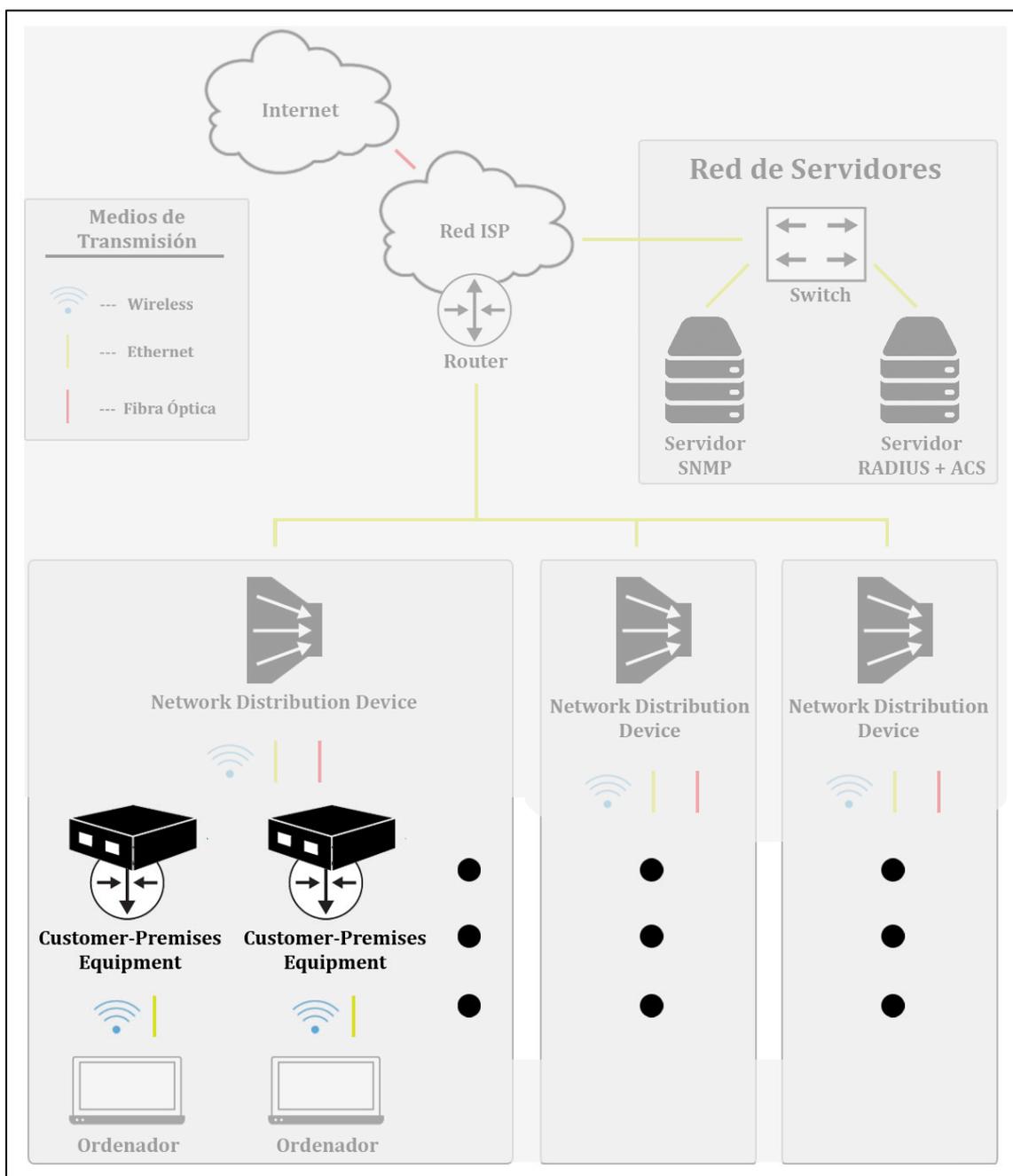


Figura 5 – Customer-Premises Equipment

Un **Equipo Local del Cliente (Customer-Premises Equipment)** antiguamente se refería al equipamiento situado en el extremo de la línea telefónica del usuario, típicamente era propiedad de la compañía telefónica.

En la actualidad, el término CPE se refiere a prácticamente cualquier dispositivo de usuario final como modem ADSL, Cablemodem, Antena Parabólica, ONT, Teléfono VoIP, etc. que puede ser tanto propiedad del usuario como del proveedor.

El equipo puede proveer una combinación de servicios ofrecidos por el proveedor incluyendo datos, voz, video y un host de aplicaciones multimedia.

Un ejemplo sería el **triple play** de un operador, donde cada uno de los servicios se entrega en una **VLAN** (Red de Área Local Virtual) distinta.

Ejemplo Telefónica en redes FTTH:

- VLAN 2 **TV**
- VLAN 3 **Voz**
- VLAN 6 **Datos**

2.1.1. Tecnologías de Última Milla

La parte de las redes que conecta los usuarios finales (residenciales o corporativos) a las redes de los operadores de telecomunicaciones se conoce como **red de acceso**, aunque también está muy extendida la denominación “**última milla**”.

Todas las conexiones entre los usuarios y las centrales forman la llamada **red de acceso**. Mientras que las conexiones entre las diferentes centrales de diferente jerarquía forman lo que se conoce como red de transporte o **red de distribución**.

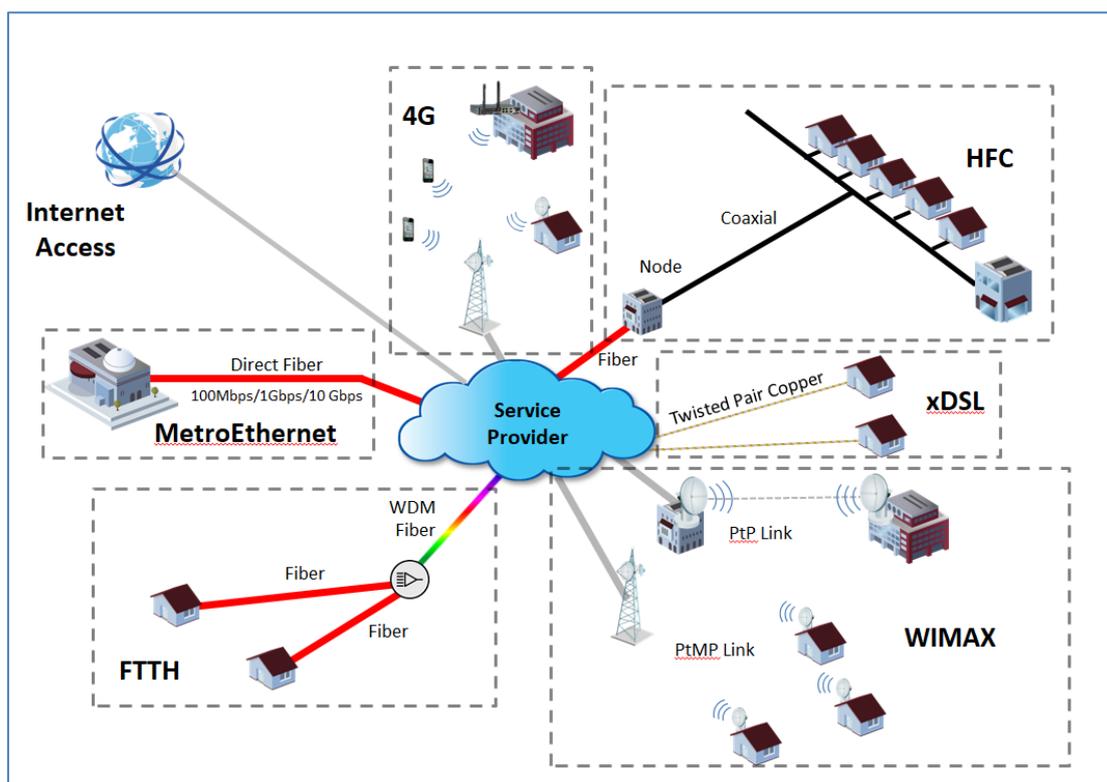


Figura 6 – Tecnologías de Última Milla

Algunas de las tecnologías existentes para conectar los usuarios finales de una red de comunicaciones con la propia red de transporte son: **Wireless/Wimax** (Redes WISP), **GPON** (FTTH), **DOCSIS** (Redes de cable coaxial), **xDSL** (Par Telefónica), etc.

WISP (Wireless ISP)

Un proveedor de servicios de Internet inalámbrico (**WISP**) es un ISP con una red basada en conexiones inalámbricas. Tienen un mercado amplio en entornos rurales donde no exista la posibilidad de realizar despliegues de infraestructura cableada.

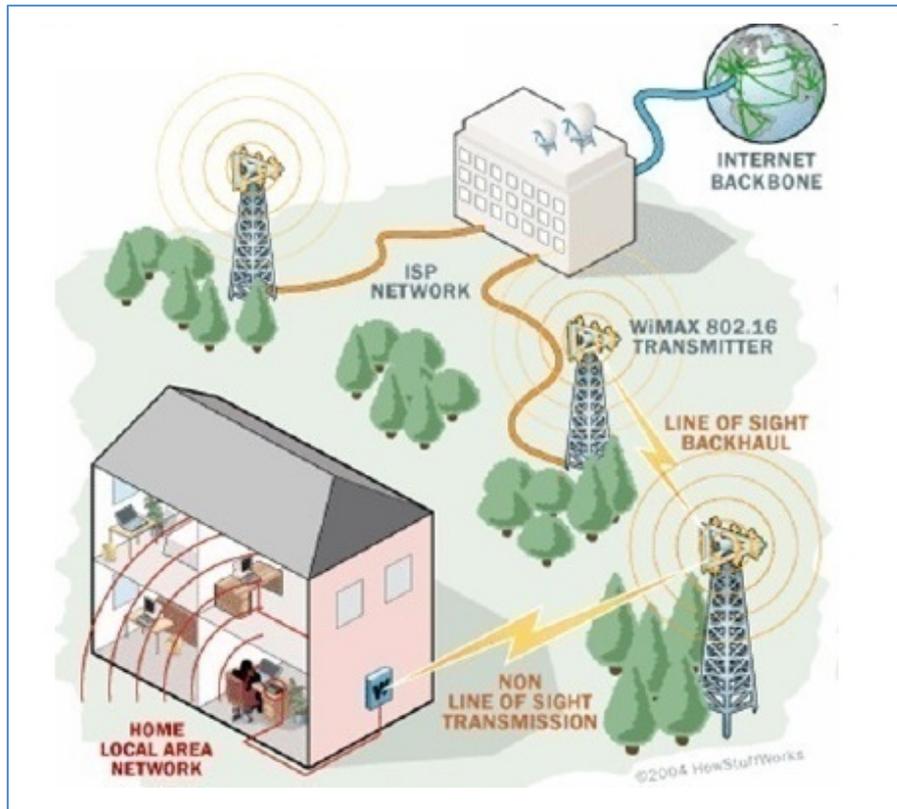


Figura 7 – Ejemplo de una red WISP

La estructura de una red **WISP**, basándonos en la capa de acceso, suele ser un conjunto de **antenas sectoriales**, un tipo de antena utilizado por los ISP para poder distribuir los servicios que ofrecen a sus clientes que abarcan gran distancia, a las que se conectan **antenas CPE**, antenas con un grado de arco más direccional con mayor potencia.

Entre antena Sectorial y antena CPE existe un **radioenlace**, la transmisión de datos o voz a través de radiofrecuencias con longitudes de onda en la región de frecuencias de microondas, de manera que se puedan comunicar ambas antenas de manera **wireless**, pudiendo abarcar distancias largas sin hacer falta de ningún tipo de cableado.

Los WISPs aparte de ofrecer Internet también **ofrecen servicios** como VPNs (Virtual Private Network), VoIP (Voice over IP), LBS (Location Based Services) y Televisión.

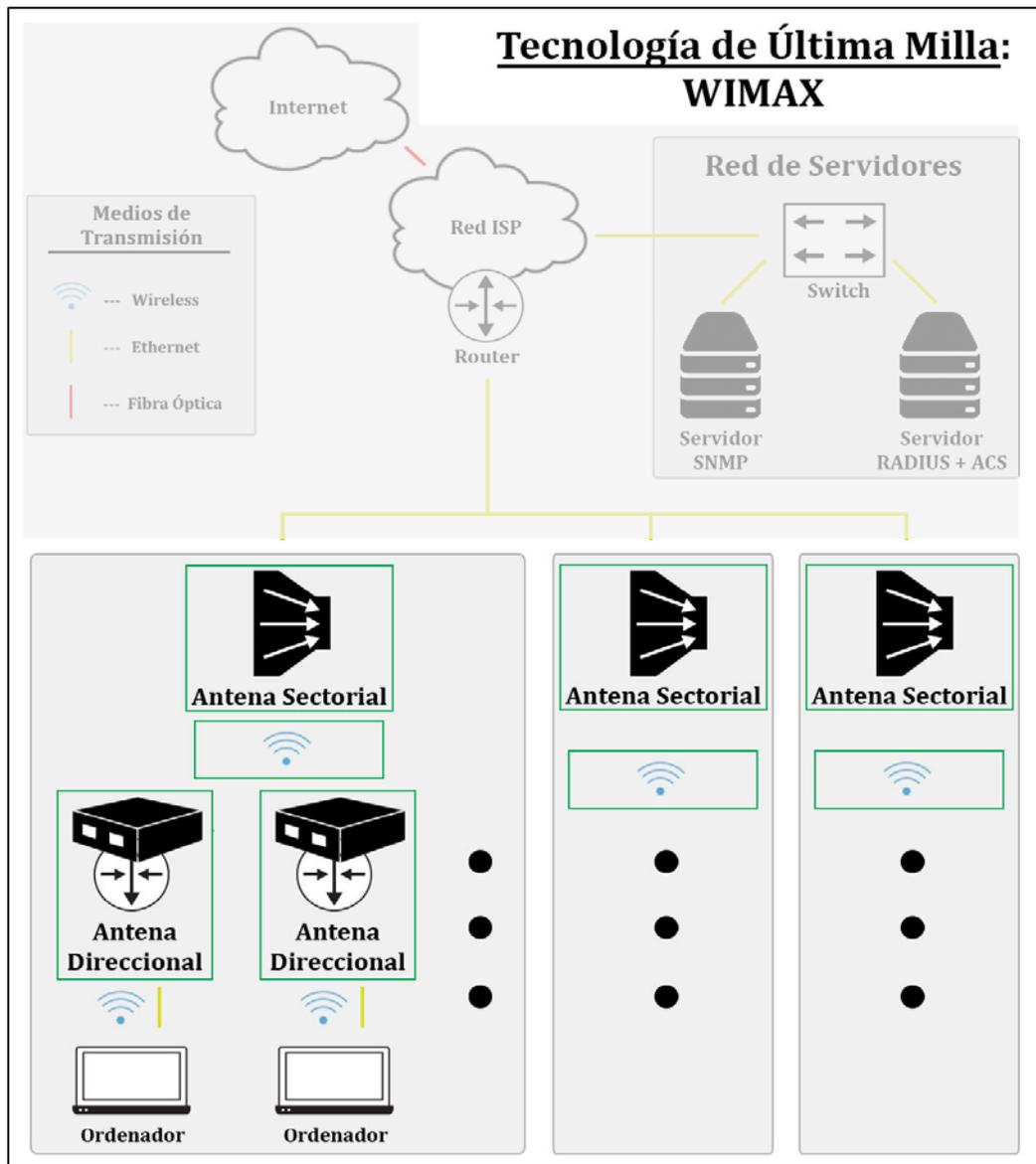


Figura 8 – Esquema típico de una red WISP

Network Distribution Device

El **Network Distribution Device** de una red WISP es una **antena sectorial**, una clase de antena de tipo sector (mirando el arco de una circunferencia, los grados del arco son: 60°, 90°, 120° en los casos más típicos). Se suelen utilizar para distancias de alcance limitado de alrededor de 6 a 8 Km.

Las antenas sectoriales, al ser antenas de tipo direccional, alcanzan **más distancia** pero con una **cobertura más baja**.

Se utilizan las antenas sectoriales para las conexiones **Point-to-Multi-Point (PtMP)**, por el hecho de que alcanzan **largas distancias** con una buena cobertura.



Figura 9 - Antena Sectorial

Es a estas antenas sectoriales donde se conectan todas nuestras antenas CPE.

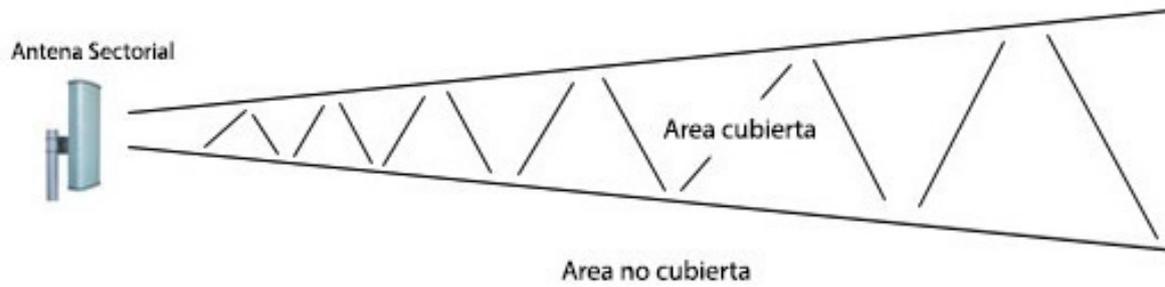


Figura 10 - Área cubierta por una Antena Sectorial

Los proveedores de servicios para montar estas antenas sectoriales utilizan el concepto **Distributed Antenna Systems (DAS)**. Es un grupo de antenas conectadas al mismo transmisor y que distribuye la señal por diferentes zonas donde no llegaría teniendo una sola antena. Este concepto se suele utilizar más en ciudades densas donde necesitan cobertura inalámbrica en todos sitios y en situaciones donde haya muchos usuarios que con una sola antena sería incapaz de gestionar todas las conexiones.

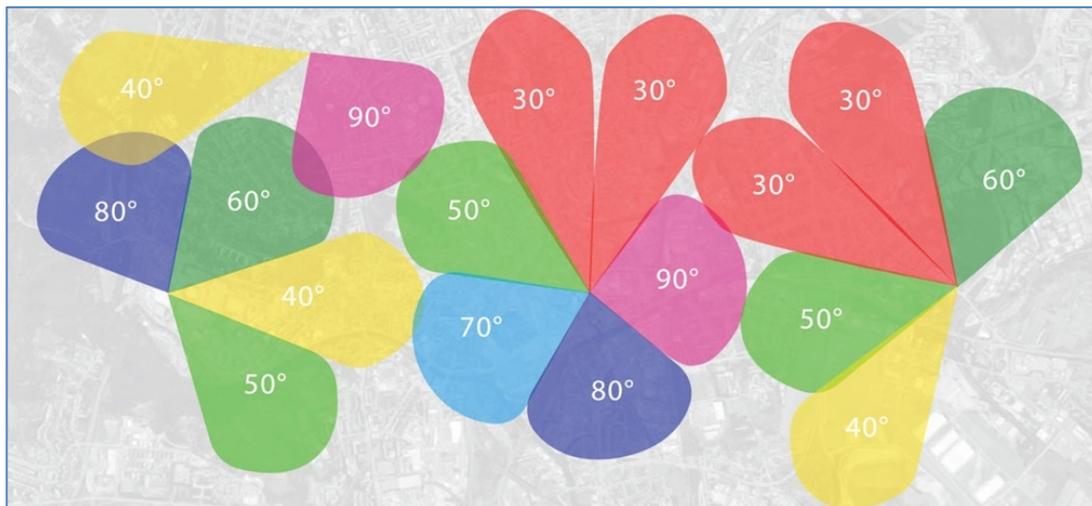


Figura 11 - Ejemplo de sistema DAS

Dependiendo de cómo están distribuidos los clientes, un ISP podría utilizar antenas sectoriales de más o menos grados del arco.

Teniendo un arco más pequeño haría que la señal emitida pudiera llegar más lejos. Estas antenas se utilizan mucho para los clientes/usuarios finales que se encuentran muy lejanos a dicha antena.



Figura 12 - Antenas Sectoriales montado DAS

Customer-Premises Equipment

En las redes WISP el dispositivo CPE es una antena, más comúnmente una **antena direccional**, que se conecta con una antena sectorial del WISP, mediante un **Point-to-Point (PtP)**, un protocolo utilizado para establecer una conexión entre dos dispositivos, eligiendo, siempre cuando se pueda, la antena sectorial con mejor señal.



Figura 13 - Antena Direccional Ubiquiti



Figura 14 - Antena Direccional Mikrotik

Las antenas direccionales, debido a que tienen un **largo alcance**, su **cobertura** es muy baja. Esto es muy importante para las antenas CPE ya que son dispositivos que se van a conectar de manera **PtP** y por tanto queremos la mínima interferencia posible. También, al ser antenas direccionales, tienen un arco de grado más cerrado, haciendo que la señal se emita de manera que evite más fácilmente el **ruido** que generan otros dispositivos que puedan emitir señales.

FTTH (Fiber to the Home)

FTTH es una tecnología de telecomunicaciones que viene del termino **FTTX** (termino genérico para designar cualquier acceso de banda ancha sobre **fibra óptica**).

En **FTTH**, la fibra óptica llega hasta el interior de las **instalaciones del cliente**. Una vez la fibra ha llegado al usuario final, la señal se puede **convertir** en cualquier medio: par trenzado, cable coaxial, wireless o fibra óptica.

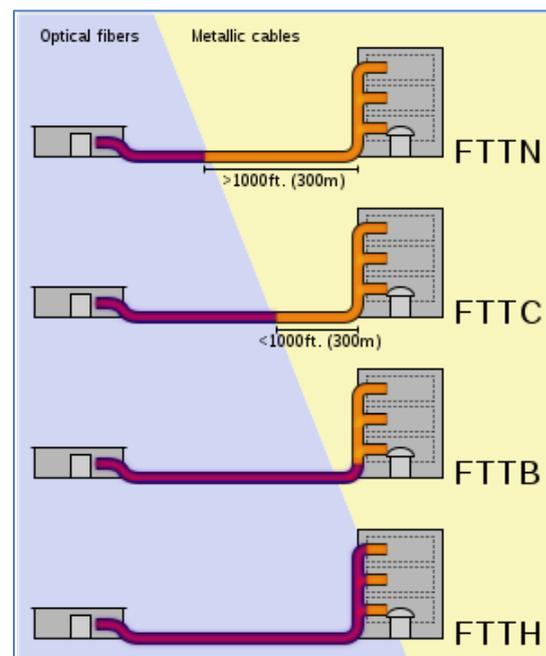


Figura 15 – Esquema Arquitectura FTTX

La estructura de una red FTTH se compone de:

- Un **Optical Line Terminal (OLT)**, un dispositivo que sirve para **convertir las señales** de luz en señales eléctricas y vice-versa y **aprovisionar** datos a los dispositivos cliente mediante un protocolo propio.
- El dispositivo cliente es un **Optical Network Terminal (ONT)**, un dispositivo que se utiliza para **desmultiplexar los servicios** ofrecidos por el ISP.

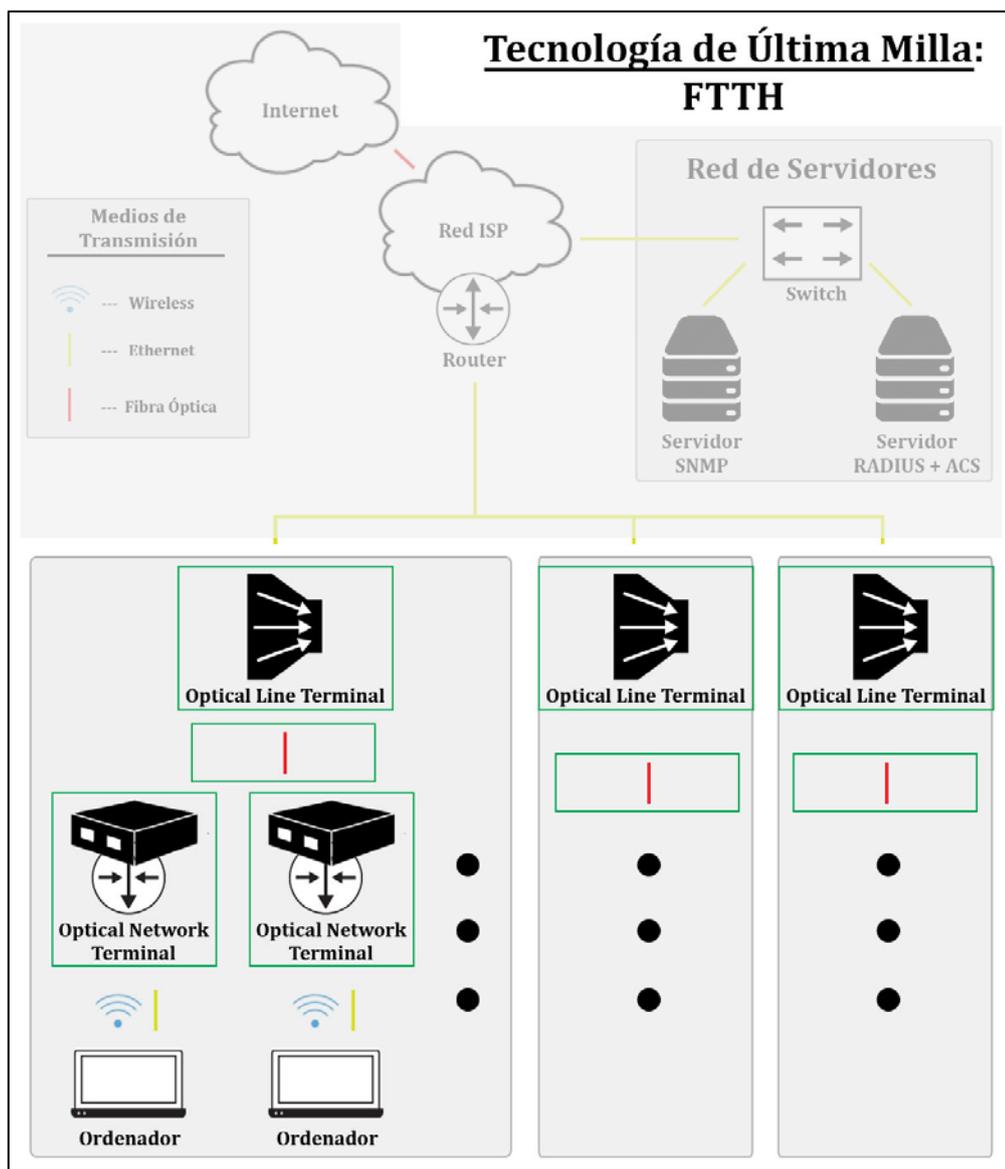


Figura 16 – Esquema típico de una Red FTTH

Entre una OLT y un ONT existe un componente llamado **Splitter**, un elemento pasivo que divide la señal de **fibra óptica** en 2, 4, 8 o más ramas. En una red FTTH, entre la OLT y una ONT, la señal típicamente se divide dos veces. La primera división con un **Splitter 1x4**, divide la señal de una fibra en 4, y la segunda división con un **Splitter 1x8**, divide la señal en 8.

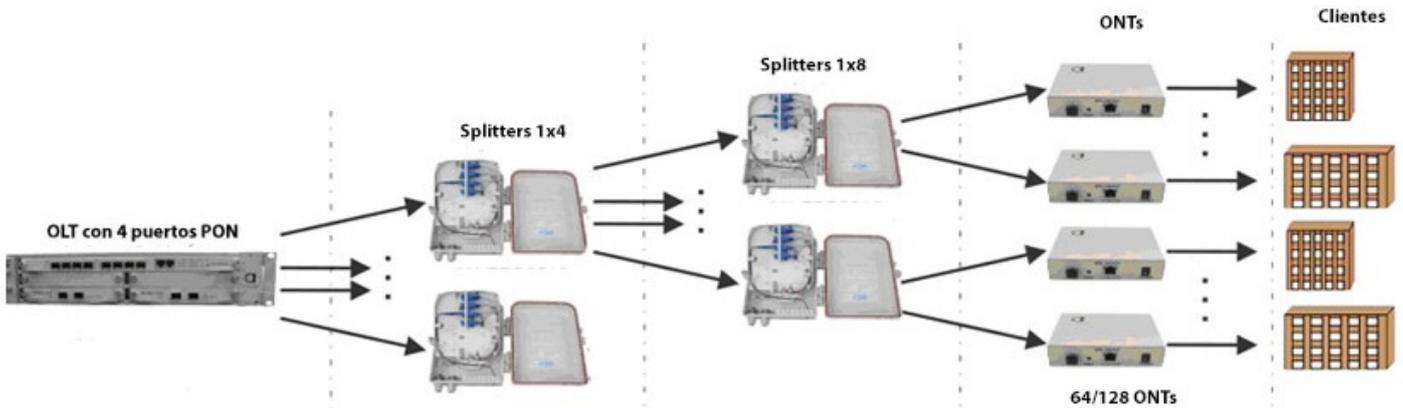


Figura 17 - Ejemplo Red FTTH

Network Distribution Device

Una OLT, “**Optical Line Terminal**”, es un dispositivo que sirve como el punto final de una red óptica pasiva del proveedor. Cumple dos funciones fundamentales:

1. La **conversión de señales** eléctricas a señales de luz, utilizada por la fibra óptica y vice-versa.
2. El **aprovisionamiento** de las ONTs conectadas a la OLT a través de Splitters (uno de los elementos pasivos más importantes en las redes FTTH que sirve para dividir la señal fibra óptica en 8 o más ramas).

El aprovisionamiento que realiza la OLT utiliza el protocolo llamado **ONU Management and Control Interface (OMCI o G.988)**, una interfaz de gestión y control para unidades de red FTTH. Especifica la configuración, la gestión de fallos y rendimiento para los dispositivos ONT.



Figura 18 - Ejemplo de OLT

Customer-Premises Equipment

ONT, “**Optical Network Terminal**”, también conocido como ONU, “**Optical Network Unit**”, es un dispositivo con funcionalidad de terminar la conexión pasiva con la OLT y demultiplexar los servicios ofrecidos por el ISP (Televisión, VoIP y acceso al Internet).

En caso de Redes FTTH la ONT convierte las señales de luz en señales eléctricas para después transmitir las por cable coaxial, par trenzado, wireless, etc.

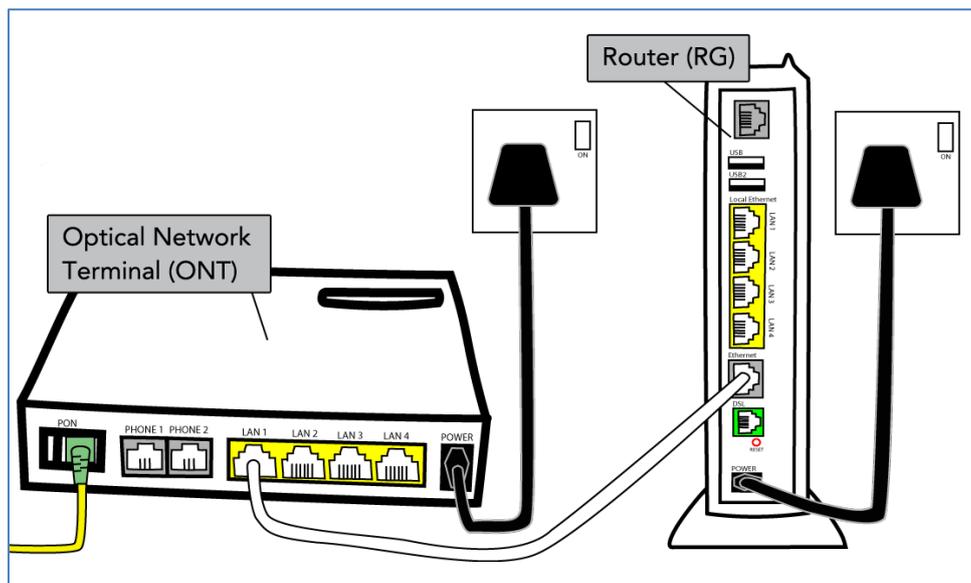


Figura 19 - Ejemplo conexasión de ONT

Hay **diferentes tipos de ONTs**: ONTs que simplemente sirven para **transformar las señales** y ONTs que también llevan la **funcionalidad de Router**, pudiendo transformar las señales emitidas y recibidas, además de funcionar como un router de casa, sin necesidad de una ONT para transformar las señales y un router conectado a ella.



Figura 20 - ONT 1-Port GPON Terminal



Figura 21 - ONT Router GPON 11ac

Otras Tecnologías (xDSL, Redes de Cable y Satélite)

Algunas otras tecnologías que se utilizan en las redes ISP son:

- **xDSL**: es una tecnología de comunicación de datos que permite una transmisión de datos más rápida sobre **líneas telefónicas de cobre** que un modem de banda vocal convencional.

En una red xDSL, como ejemplo el **ADSL**, el **Network Distribution Device** se conoce como **Digital-Subscriber-Line-Access-Multiplexer (DSLAM)** y el **Customer-Premises Equipment** se conoce como **Modem ADSL**. Estos dispositivos están conectados mediante un cable **Par Trenzado**.

- **Redes de Cable**: red de comunicaciones que utiliza cableado de **fibra óptica** en la red de distribución y **cable coaxial** en la red de acceso. El acceso utilizando esta tecnología se conoce simplemente como "cable", por ejemplo: "Tengo dos posibilidades para contratar el acceso a Internet desde mi casa, por ADSL o por cable". También se conoce como **Hybrid Fiber-Coaxial (HFC)**.

En las Redes de Cable, el **Network Distribution Device** se conoce como **Cable Modem Termination Systems (CMTS)** y el **Customer-Premises Equipment** se conoce como **CableModem**. Es una red compuesta por **Fibra Óptica** y **Cable Coaxial**.

- **Satélite**: el acceso a Internet mediante un satélite es la única opción viable en algunas zonas, especialmente zonas rurales, montañosas o de **difícil acceso** donde no existe tendido de cable ni cobertura 3G. Para utilizar un acceso por satélite es necesario el uso de una **antena parabólica** y de un modem DVB-S específico para este tipo de conexiones que permite comunicación bidireccional.

En las redes Satélite, el **Network Distribution Device** es un **Centro de control de satélites** que gestiona los datos enviados y recibidos de los satélites y el **Customer-Premises Equipment** es una **antena** enfocada al satélite. Estos dispositivos están conectados mediante un **radioenlace**.

2.1.2.AAA (Authentication, Authorization, and Accounting)

"**Autenticación, Autorización y Registro**", es un **modelo de seguridad** que se utiliza para poder identificar a usuarios y controlar el acceso de los mismos a diferentes recursos en un sistema informático, y a la vez registrando como se utiliza cada recurso.

Se trata de habilitar la movilidad de los usuarios y conseguir que los parámetros de seguridad puedan variarse muy fácilmente con simples configuraciones. Sin AAA, una red puede estar configurada estáticamente para controlar el acceso (las direcciones IP estáticas, los sistemas no se pueden mover y las opciones de conectividad bien definidas).

Algunos protocolos que implementan el modelo de seguridad AAA son **RADIUS** (Remote Authentication Dial-In User Service), un protocolo para gestionar los usuarios y su uso de los servicios ofrecidos por el ISP, y **TACACS+** (Terminal Access Controller Access-Control System Plus), un protocolo desarrollado por CISCO.

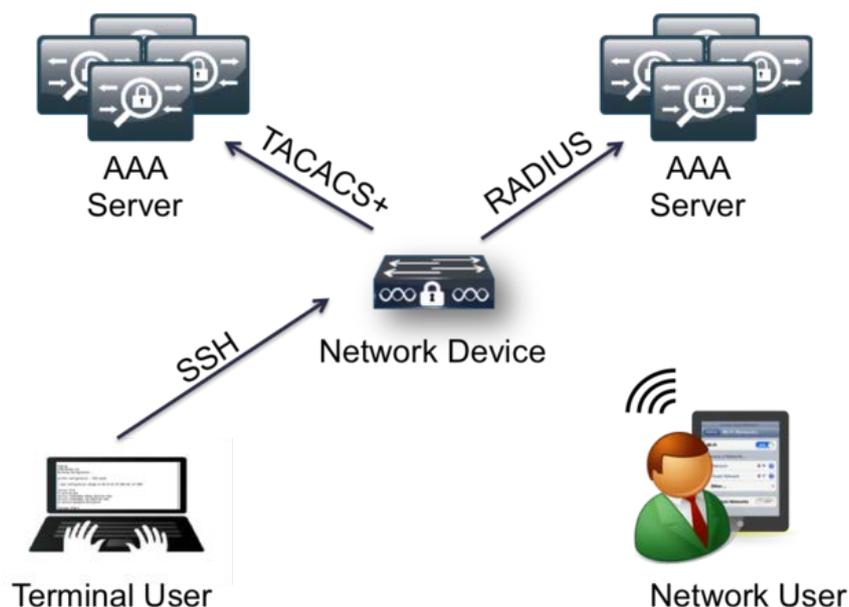


Figura 22 – Diseño AAA

RADIUS

RADIUS es un protocolo de red que permite gestionar los usuarios y su uso de los servicios ofrecidos por el ISP. Cuando se realiza la conexión con un ISP, se envía una información que generalmente es un nombre de usuario y una contraseña.

En el caso de un ISP, esta información se transfiere a un router típicamente sobre el protocolo **PPP**, un protocolo utilizado para establecer una conexión directa entre dos nodos de una red, el cual redirige la petición a un servidor RADIUS sobre el protocolo RADIUS.

El servidor RADIUS comprueba que la información es correcta utilizando esquemas de autenticación como **PEAP** o **EAP**. Si es aceptado, el servidor autorizará el acceso al sistema del ISP y el router le asignará los recursos de red como una dirección IP.

El servidor RADIUS puede tener incorporado una **base de datos** para guardar los datos de conexión de los usuarios y gestionar permisos, o puede utilizar una base de datos externa.

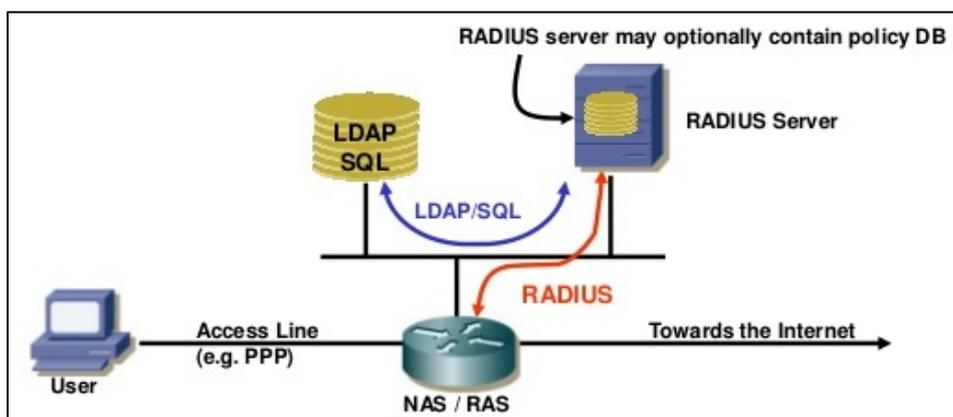


Figura 23 – Configuración típica RADIUS

Aprovisionamiento de direccionamiento IP

El aprovisionamiento de direcciones IP se basa en un proceso en el que un dispositivo servidor (hoy en día esta función de servidor suele estar implementado en el propio router) aporte a un dispositivo cliente una dirección IP que puede ser tanto estática como dinámica.

DHCP

DHCP, “**Dynamic Host Configuration Protocol**” es un protocolo de administración de red utilizado en redes TCP/IP mediante el cual un servidor DHCP asigna dinámicamente una dirección IP y otros parámetros de configuración de red, **DHCP Options**, a cada dispositivo de una red para que puedan comunicarse con otras redes IP.

Además de ser una gran comodidad, DHCP garantiza la estabilidad de varias maneras: **renovación periódica y reenlace**. Si el servidor DHCP se ha caído por algún motivo, el cliente repetirá la solicitud hasta que el servidor le responda con un direccionamiento IP. En cuanto a la seguridad, sin embargo, DHCP no está tan bien pensado. No contiene ningún mecanismo para la autenticación, no implementa AAA. Esto lo hace vulnerable a los ataques.

Opción 43

Vendor Specific Information, también conocido como **Opción 43**, es una opción que se utiliza en el **protocolo DHCP**. Se utiliza para aportar a dispositivos clientes conectados a una red la ubicación donde se encuentra un servidor ACS.

El dispositivo cliente CPE inicia la comunicación pidiendo una dirección IP al servidor DHCP con un **DHCP REQUEST**. En este DHCP REQUEST, el CPE especifica al servidor DHCP si soporta el uso de **TR-069 (CWMP)** mediante el uso de otras opciones (opción 60 y 124).

El servidor DHCP responde con un **DHCP OFFER**, y aquí dentro especifica al CPE la dirección del servidor ACS mediante el uso de la **opción 43**.



Figura 24 - DHCP OFFER devolviendo la dirección del Servidor ACS

Opción 66

Trivial File Transfer Protocol (TFTP) Server Name, también conocido como **Opción 66**, es una opción que se utiliza en el **protocolo DHCP**.

Se utiliza para aportar a dispositivos clientes conectados a una red la ubicación donde se encuentra un **servidor TFTP**, servidor que permite a un cliente descargar o subir un fichero de tamaño pequeño mediante el **protocolo TFTP**.

Estos ficheros suelen ser **ficheros de configuración** que se aportan a los dispositivos CPE para cargar dicha configuración.

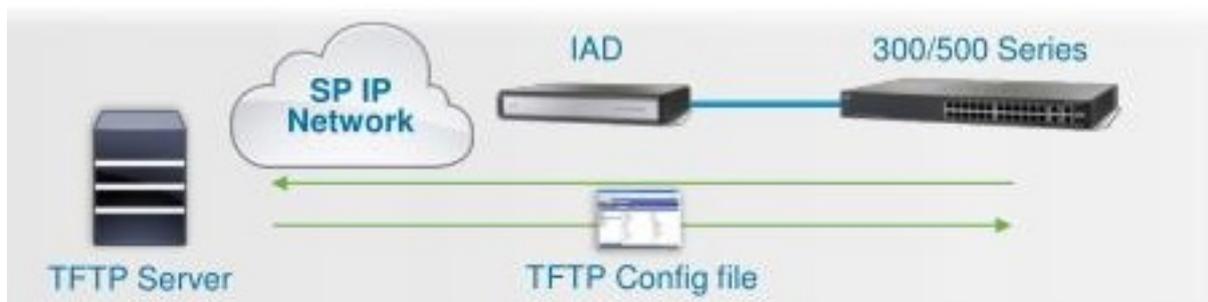


Figura 25 – Petición y envío de un archivo de configuración

PPP

PPP, "**Point-to-Point Protocol**", es un protocolo del nivel de enlace de datos (Capa 2), utilizado, más típicamente, para establecer una **conexión directa entre dos puntos** de una red. Conecta dos Routers directamente sin falta de ningún dispositivo extra entre medias de ambos. Aprovechando direccionamiento IP a dispositivos cliente pudiendo hacer uso de AAA.

PPPoE

PPPoE, “**Point-to-Point Protocol over Ethernet**”, es un protocolo de red para la encapsulación PPP sobre una capa de Ethernet que utiliza métodos estándares de cifrado, autenticación y compresión especificados por PPP.

Es ampliamente utilizado por los ISP para aprovisionar los servicios de Internet utilizando el concepto de **AAA** para autenticar usuarios (**A**uthentication), especificarles perfiles con unos permisos/restricciones específicos (**A**uthorization) y monitorizar su actividad en la red (**A**ccounting).

Cuando se **inicia** una sesión PPPoE, se le **asigna** una dirección IP. Una vez se **cierra** la sesión, la dirección IP se **libera**, lo que permite una **reutilización** eficiente de las direcciones IP.

AAA

Authentication: es el proceso que determina si el cliente (un usuario, dispositivo o proceso software) es un usuario legítimo del sistema. Se consigue mediante la presentación de una propuesta de identidad (ej. **Nombre de usuario**) y la demostración de estar en posesión de credenciales que permiten comprobarla (ej. **Contraseña o Certificados Digitales**). Una vez autenticada se pasa a la Autorización.

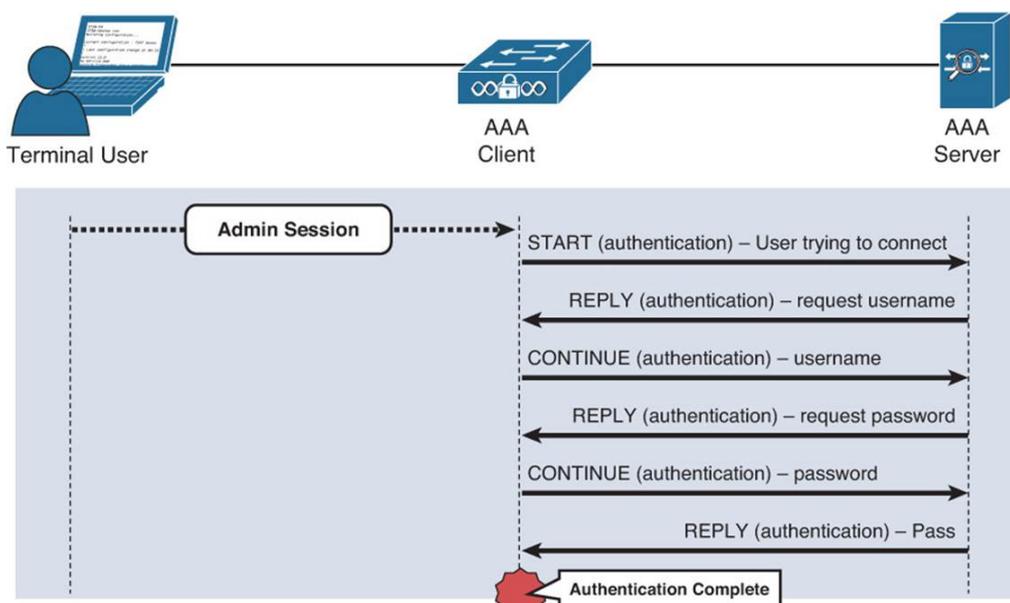


Figura 26 - Mensajes de Autenticación entre cliente y servidor

Authorization: es el proceso que determina lo que un cliente autenticado (un usuario, dispositivo o proceso software) pueda hacer en la red. Se basa en los **privilegios o restricciones** que dicho cliente autenticado pueda tener. La mayor parte de las veces el privilegio concedido consiste en el uso de un determinado tipo de servicio (ej. Filtrado de direcciones IP, asignación de Ancho de Banda).

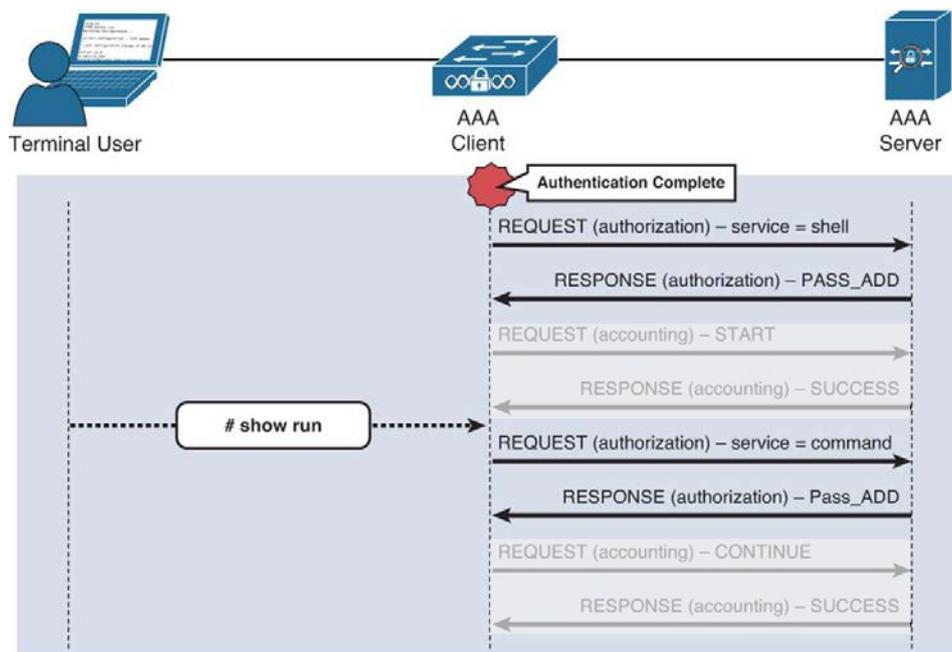


Figura 27 - Mensajes de Autorización entre cliente y servidor

Accounting: es el proceso de **monitorizar** y grabar el uso de la red del cliente. Esta información puede usarse posteriormente para la administración, planificación, facturación, etc. La información típica que un proceso de contabilización registra es la identidad del usuario, el tipo de servicio que se le proporciona, cuando comenzó a usarlo, y cuando terminó. En muchos ISPs también se registran los **Bytes transferidos** y el **tiempo activo** con el objetivo de restringir al usuario si sobrepasa una cantidad de Bytes o un tiempo especificado.

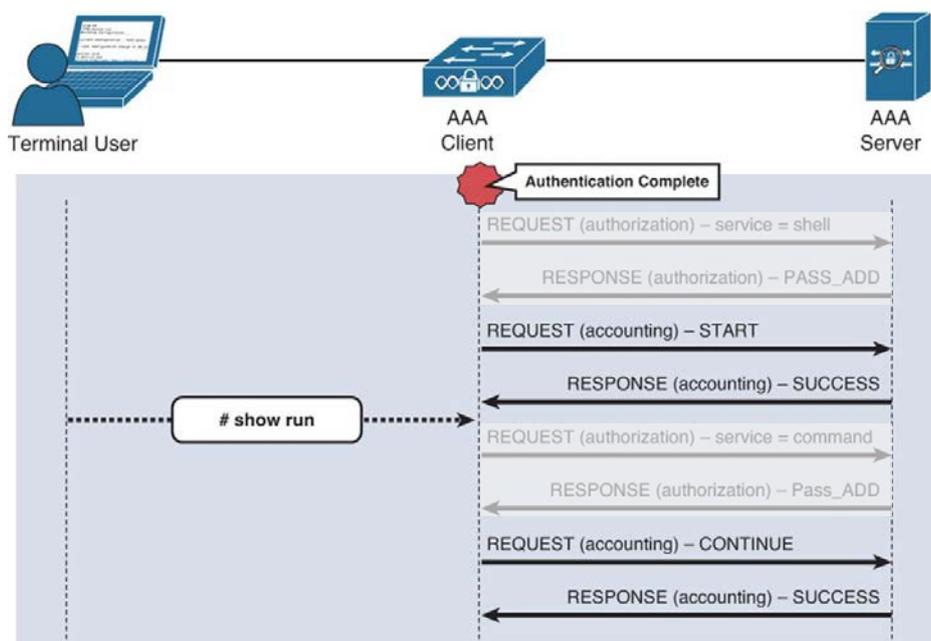


Figura 28 - Mensajes de Registro entre cliente y servidor

2.2. Auto-aprovisionamiento

Hay dos tipos de aprovisionamiento: auto y manual. **Auto-aprovisionamiento** permite al ISP desplegar servicios instantáneamente mediante el uso de una aplicación software, mientras que con el **aprovisionamiento manual**, el técnico tiene que configurar todos los dispositivos uno por uno.

El **auto-aprovisionamiento** es muy útil cuando el ISP dispone de un número elevado de clientes y tiene que realizar una configuración avanzada a los dispositivos.

La comunicación entre un **CPE** y un **Servidor de configuración automática (ACS)**, se realiza mediante el protocolo CPE WAN Management Protocol (**CWMP** o **TR-069**). Las peticiones enviadas entre CPE y servidor ACS se basan en el protocolo Simple Object Access Protocol (**SOAP**), un protocolo estándar de mensajería, y **TR-098**, un estándar que especifica el modelo de datos utilizado por todos los CPE.

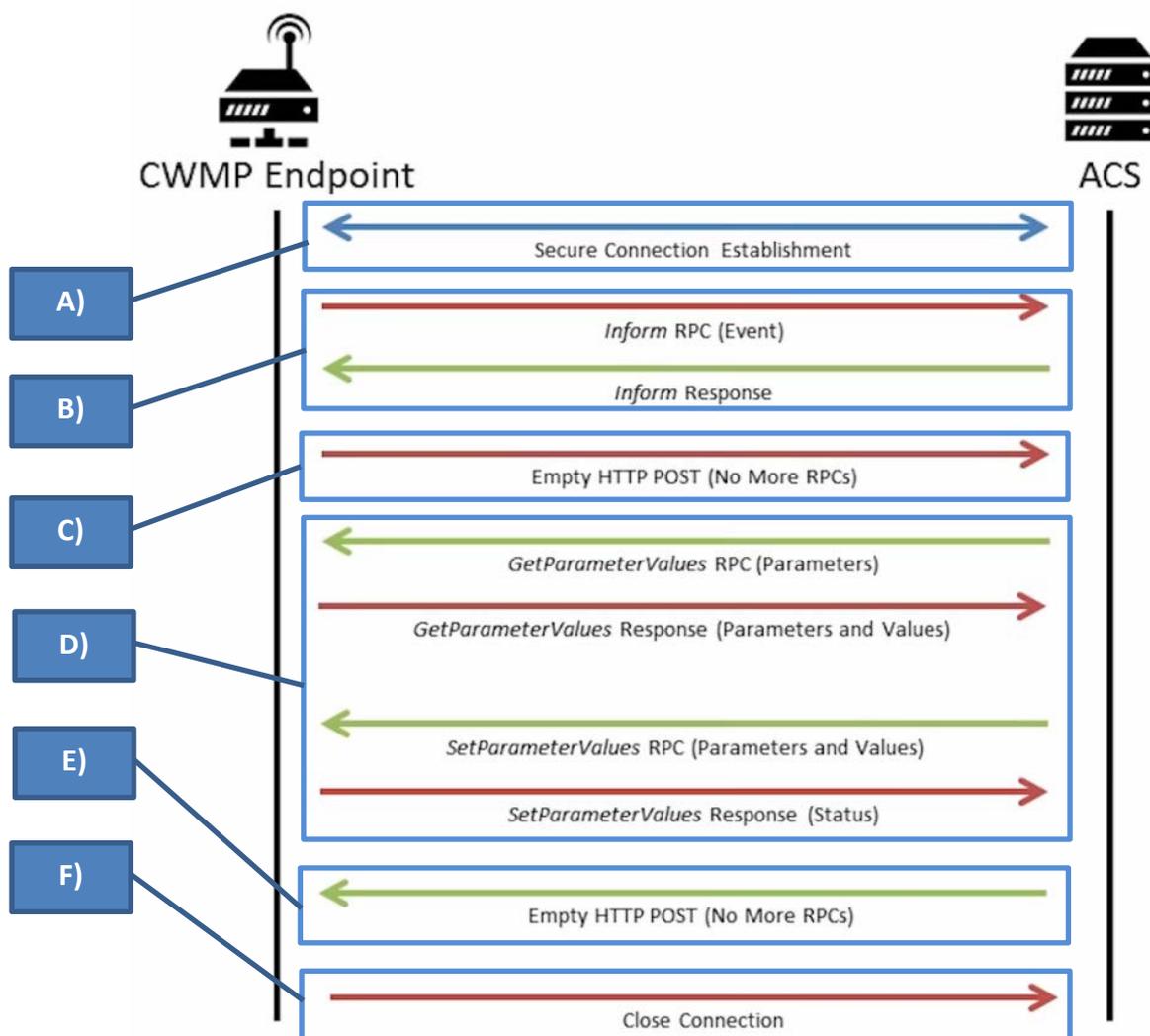


Figura 29 - Ejemplo de comunicación entre CPE y ACS

Un ejemplo de comunicaciones entre CPE y servidor ACS sería:

- A. Si es la primera comunicación entre el CPE y el ACS, el CPE realiza la **primera petición** para crear una conexión. Estas comunicaciones pueden utilizar **Secure Socket Layer (SSL)** opcionalmente, para que la comunicación sea **segura**.
- B. El CPE envía al servidor ACS un **Inform**, una función que indica el estado del CPE al ACS. Sin embargo, si el CPE ya está registrado con el ACS, el ACS puede enviar una petición para volver a empezar una conversación. El ACS responde a esta petición, informando al CPE de que la petición ha sido **aceptada**.
- C. El siguiente mensaje que manda el CPE es una **petición vacía** para informar al servidor ACS de que la sesión sigue abierta, para poder **empezar** a enviar funciones al CPE.
- D. El servidor ACS, una vez recibida la petición vacía, empieza a enviar funciones **Remote Procedure Call (RPC)**, operaciones entre un ACS y un CPE. En esta fase se envía la configuración al CPE mediante las funciones RPC, algunos ejemplos de estas funciones son: **GetParameterValue** (Obtener el valor de un parámetro del CPE) y **SetParameterValue** (Asignar el valor de un parámetro del CPE). El CPE responde a las funciones RPC del servidor ACS con una **respuesta**.
- E. El servidor ACS envía una **petición vacía** para indicar al CPE que ha **terminado** de enviar las funciones RPC.
- F. Una vez recibida la petición vacía, el CPE envía una petición **“Cerrar conexión”** al servidor ACS.

2.2.1. CWMP

CWMP, **“CPE WAN Management Protocol”**, también conocido como **TR-069** es un estándar que define un protocolo de capa de aplicación para la administración remota de los CPE.

Define las funciones de soporte para el auto-aprovisionamiento, la gestión de aprovisionamiento de firmware y software, gestión de dispositivos, estado y rendimiento de dispositivos, y diagnósticos.

CWMP es un **protocolo basado en texto**, los pedidos enviados entre CPE (Cliente) y ACS (Servidor) se transportan a través de HTTP (o más frecuentemente HTTPS). Estas sesiones de peticiones las suelen iniciar el CPE y empieza con un mensaje **Inform**. El servidor responde típicamente con un mensaje **InformResponse**, con ello finalizando la etapa de inicialización.

2.2.2. ACS

Un ACS, **Auto-Configuration Server**, es un servidor capaz de **gestionar dispositivos remotos** para aprovisionar configuraciones, actualizaciones de software y firmware, y observar el estado de dichos dispositivos.

Existen muchas plataformas donde poder realizar estas funciones de ACS, algunas de ellas de pago como: AVSystem, Axiros y Friendly Tech. Las alternativas Open Source son: **GenieACS**, FreeACS y OpenACS.

Se puede distinguir dos partes importantes, una parte donde se realizan todas las funciones, también conocido como **Backend**, y una parte visual para facilitar la configuración de los parámetros de los dispositivos, conocido como **Frontend**.

En la capa **Backend** se encuentra:

- Una **base de datos** para poder **guardar** toda la configuración de los CPE, **presets** (ajustes establecidos por el ISP), objetos para aprovisionar a los CPE y ficheros como actualizaciones de software.
- Un **framework**, también conocido como **API**, donde se encuentran todas las funciones que pueda realizar el servicio ACS. Un servicio ACS realiza uso de este framework para **comunicarse con los CPEs**.

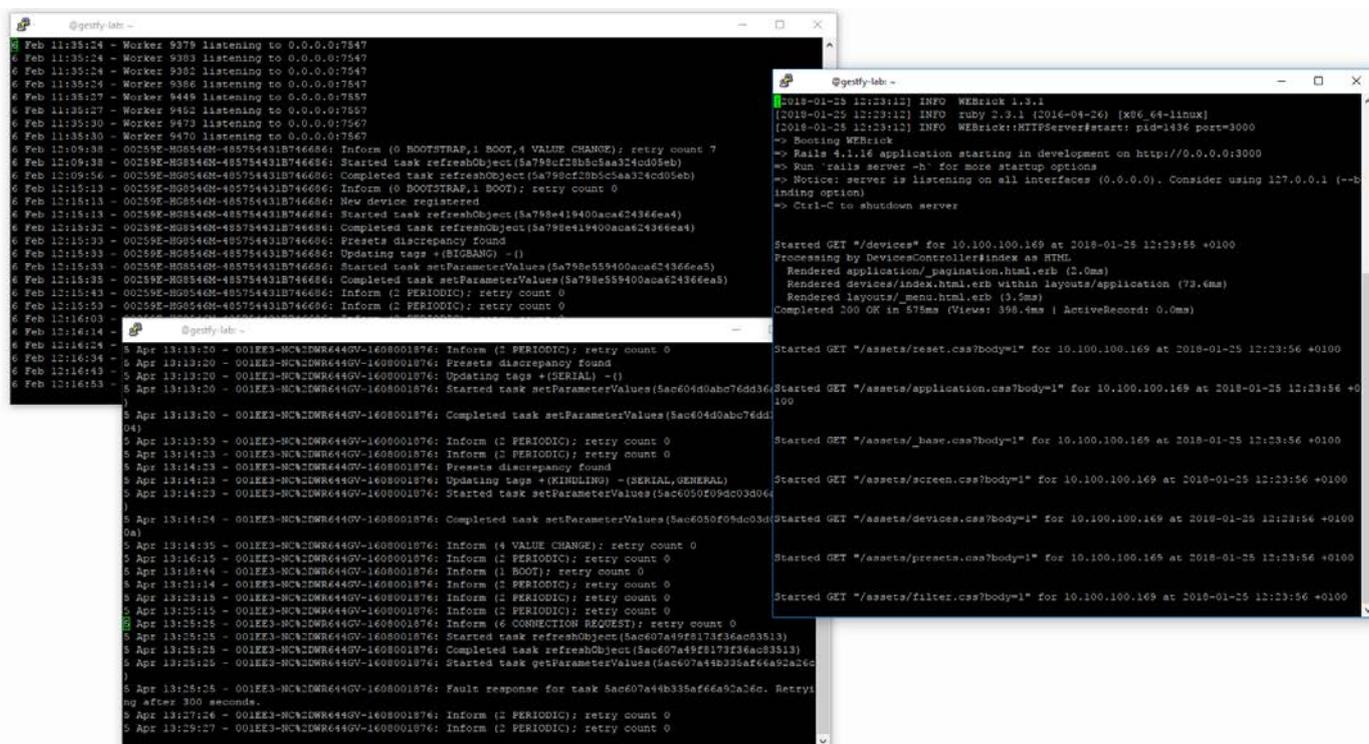


Figura 30 – Servidor ACS capa Backend

En la capa **Frontend** de cualquier plataforma ACS se suele encontrar: Una **página principal** para tener una vista simple de los dispositivos conectados al servidor ACS y su estado actual, y una página donde se demuestra la **configuración de cada dispositivo**. Allí dentro se pueden modificar los valores de los parámetros establecidos, enviar actualizaciones de software, reiniciar el dispositivo, etc.

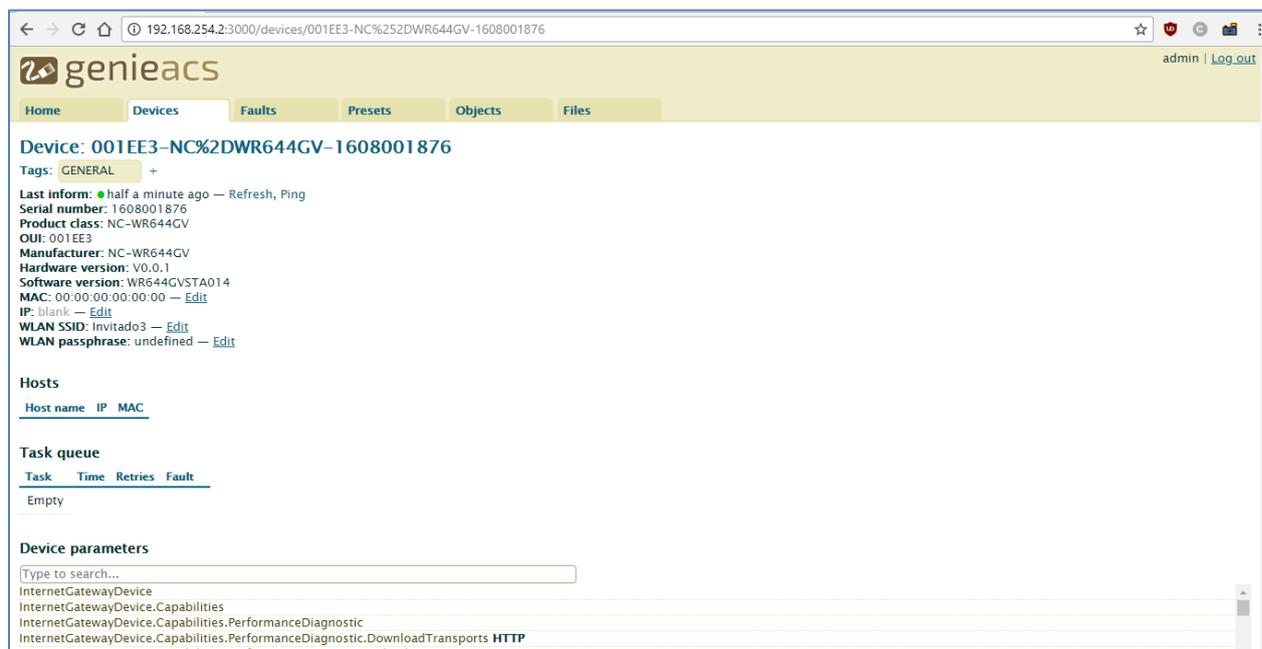


Figura 31 – Servidor ACS capa Frontend

2.2.3. SOAP

Simple Object Access Protocol, o **SOAP**, es un protocolo estándar de mensajería para el intercambio de información estructurada en redes informáticas. Utiliza **Extensible Markup Language (XML)** para su formato de mensaje, y se basa en protocolos de aplicación, más a menudo **Hypertext Transfer Protocol (HTTP)** o **Simple Mail Transfer Protocol (SMTP)**, para la negociación y transmisión de mensajes.

Este protocolo permite que los procesos que se ejecutan en sistemas operativos diferentes (como Windows y Linux) se comuniquen utilizando XML. Dado que los protocolos web como HTTP están instalados y ejecutándose en todos los sistemas operativos. SOAP facilita a los clientes invocar servicios web y recibir respuestas independientes del idioma y las plataformas.

```
<soapenv:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:cwmp="urn:dslforum-org:cwmp-1-0"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <cwmp:ID soapenv:mustUnderstand="1">1323392</cwmp:ID>
  </soapenv:Header>
  <soapenv:Body>
    <cwmp:SetParameterValues>
      <ParameterList soap:arrayType="cwmp:ParameterValueStruct[1]">
        <ParameterValueStruct>
          <Name>InternetGatewayDevice.ManagementServer.URL</Name>
          <Value xsi:type="xsd:string">http://someacs.cwmp.org/path/</Value>
        </ParameterValueStruct>
      </ParameterList>
      <ParameterKey>1323392</ParameterKey>
    </cwmp:SetParameterValues>
  </soapenv:Body>
</soapenv:Envelope>
```

Figura 32 – Estructura de un mensaje SOAP

Los elementos que destacamos de la estructura de un mensaje SOAP son:

- **<cwmp:SetParameterValue>**: Es la **función RPC** que se ha enviado mediante SOAP, en este ejemplo es la función **SetParameterValue**.
- **<Name>**: Es el nombre del **parámetro/objeto** a la que hace referencia la función RPC. En este ejemplo, se muestra como el parámetro **InternetGatewayDevice.ManagementServer.URL**.
- **<Value>**: Es el **valor enviado o recibido** al realizar la función RPC. En este ejemplo es la URL de un servidor ACS "**http://someacs.cwmp.org/path/**" en formato de un string.

2.2.4. RPC

Un **Remote Procedure Call (RPC)**, cuando hablamos de Auto-aprovisionamiento, es una operación entre un ACS y un CPE. Se utiliza para la comunicación bidireccional entre CPE y ACS basándose en el protocolo **SOAP**.

Podemos destacar 4 grupos de funciones RPC para CWMP: **Mantenimiento, Parámetros, Ficheros y Objetos**.

- **Mantenimiento**: Funciones RPC que se basan en monitorizar y gestionar el estado del dispositivo.
 - o **Inform**: El dispositivo CPE envía este mensaje al ACS para, en caso de no haber enviado ningún mensaje previamente o de no existir en la base de datos del ACS, iniciar una sesión, o en caso de estar ya registrado en la base de datos del ACS, enviar el estado actual de sus parámetros.

- **Reboot:** El ACS reinicia un CPE de manera remota cuando el CPE se encuentra con un fallo, o después de realizar una actualización de software.
- **Parámetros:** Funciones RPC que se basan en los parámetros del modelo de datos del dispositivo (**TR-098**).
 - **GetParameterValue:** Utilizado por el ACS para pedir información al CPE sobre uno o más de sus parámetros.
 - **SetParameterValue:** El ACS establece el valor de uno o más parámetros del CPE.
- **Ficheros:** Funciones RPC que se basan en los **archivos de configuración** que podemos aportar o generar de un dispositivo.
 - **Download:** Cuando el ACS requiere que un CPE descargue un fichero para poder actualizar software/firmware o un fichero de configuración.
 - **Upload:** Cuando el ACS requiere que un CPE suba un fichero a un lugar específico.
- **Objetos:** Funciones RPC que se basan en las instancias de objetos, **agrupación de parámetros**, que se pueden aportar a un CPE.
 - **Add Object:** Permite al ACS crear instancias de objetos en el CPE, como por ejemplo, entradas de Port Mapping o **conexiones WAN**.
 - **Delete Object:** Permite al ACS eliminar instancias de objetos disponibles en el CPE.

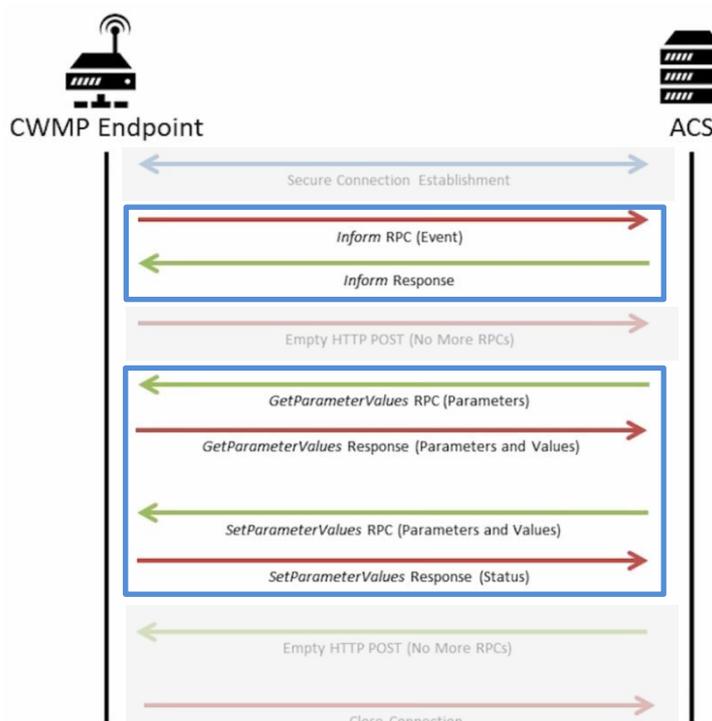


Figura 33 - Ejemplo de RPC entre CPE y ACS

2.2.5. TR-098

Es un documento realizado por **Broadband Forum** en septiembre 2005 que especifica un **modelo de datos** para el protocolo CWMP (TR-069). La estructura de este modelo de datos se basa en el uso de **Parámetros y Objetos**:

- **Parámetros**: Un par de **Nombre-Valor** que representa un parámetro de CPE accesible para un ACS para leer y/o escribir.
- **Objetos** Un **contenedor** para parámetros y/u otros objetos. El nombre completo de la ruta de un parámetro viene dado por el nombre del parámetro adjunto al nombre completo de la ruta del objeto en el que está contenido.

Algunos de los objetos y parámetros más utilizados de **InternetGatewayDevice** para CWMP son:

- **InternetGatewayDevice.DeviceInfo.:** Aquí se especifica información general sobre el dispositivo.
 - **Manufacturer**: El fabricante del CPE.
 - **ManufacturerOUI**: Identificador único de organización del CPE. Este valor se mantiene fijo siempre.
 - **SerialNumber**: Número de Serie del CPE. Este valor se mantiene fijo siempre.
 - **HardwareVersion**: Un String identificando la versión de hardware que tiene el CPE.
 - **SoftwareVersion**: Un String identificando la versión actual de software instalado en el CPE.
- **InternetGatewayDevice.ManagementServer.:** Aquí se especifican parámetros relacionadas con la asociación del CPE con el ACS.
 - **ConnectionRequestUsername**: Nombre de usuario utilizado para autenticar un ACS realizando una solicitud de conexión con el CPE.
 - **ConnectionRequestPassword**: Contraseña de usuario utilizado para autenticar un ACS realizando una solicitud de conexión con el CPE. Este valor siempre devolverá valor nulo cuando se intenta leer, asegurando la seguridad.
 - **PeriodicInformEnable**: Un booleano especificando si el CPE tiene que mandar información periódicamente su estado actual al ACS utilizando la función RPC **Inform**.
 - **PeriodicInformInterval**: La duración en segundos del intervalo de tiempo entre cada mensaje **Inform** que el CPE enviará al ACS si **PeriodicInformEnable** tiene el valor de **True**.

3. Implementación Práctica

3.1. Topología de Red

Para realizar este proyecto, empezamos primero **diseñando una red ISP** utilizando las tecnologías FTTH y WIMAX y asignando direcciones IP como en un caso real.

En la oficina montamos un **laboratorio** para realizar pruebas antes de que se implementen en la propia red ISP. Basamos nuestro laboratorio en el esquema siguiente:

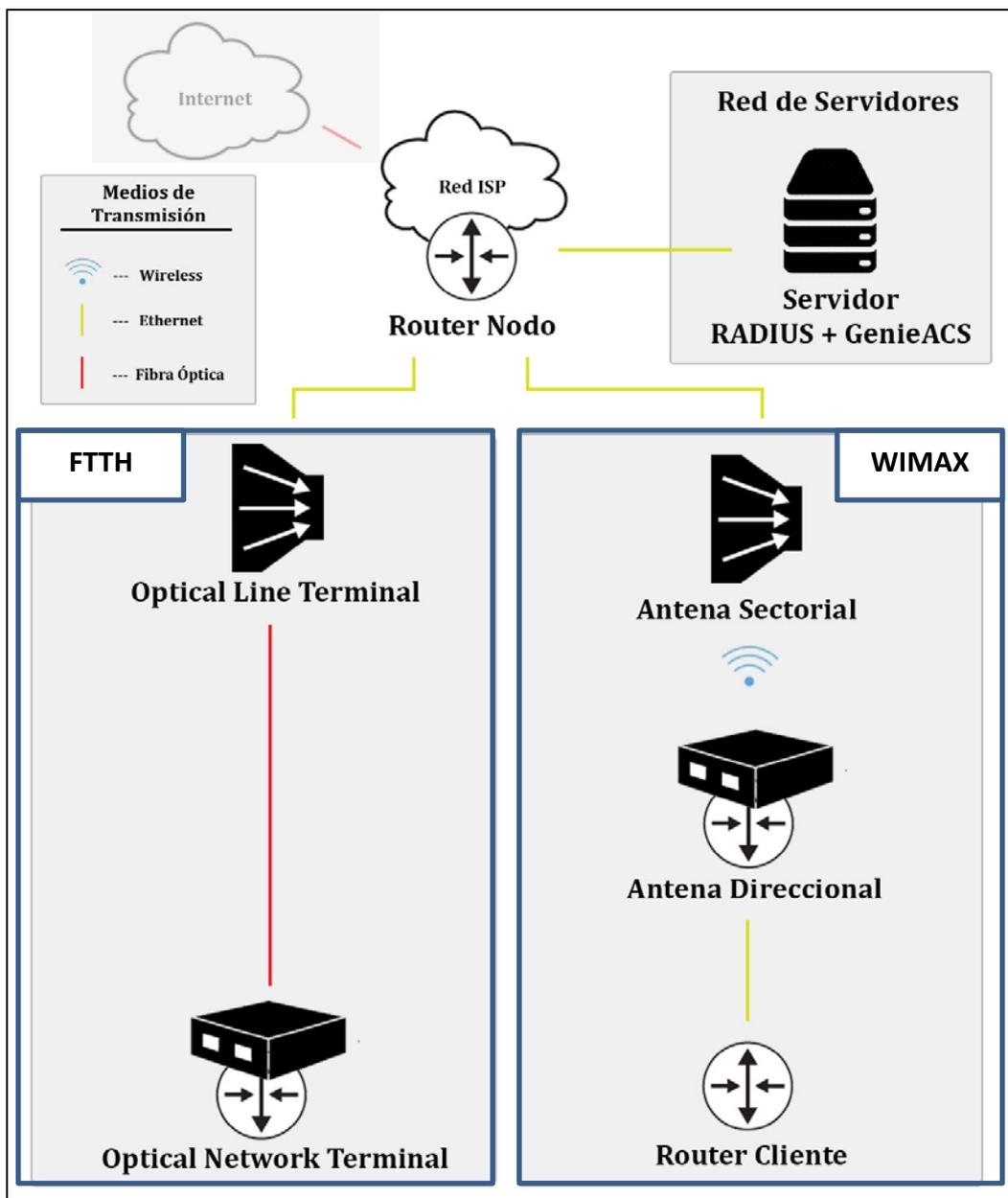


Figura 34 – Esquema de Red del Laboratorio

3.1.1. Infraestructura de Red

La infraestructura de red de nuestro proyecto es la siguiente:

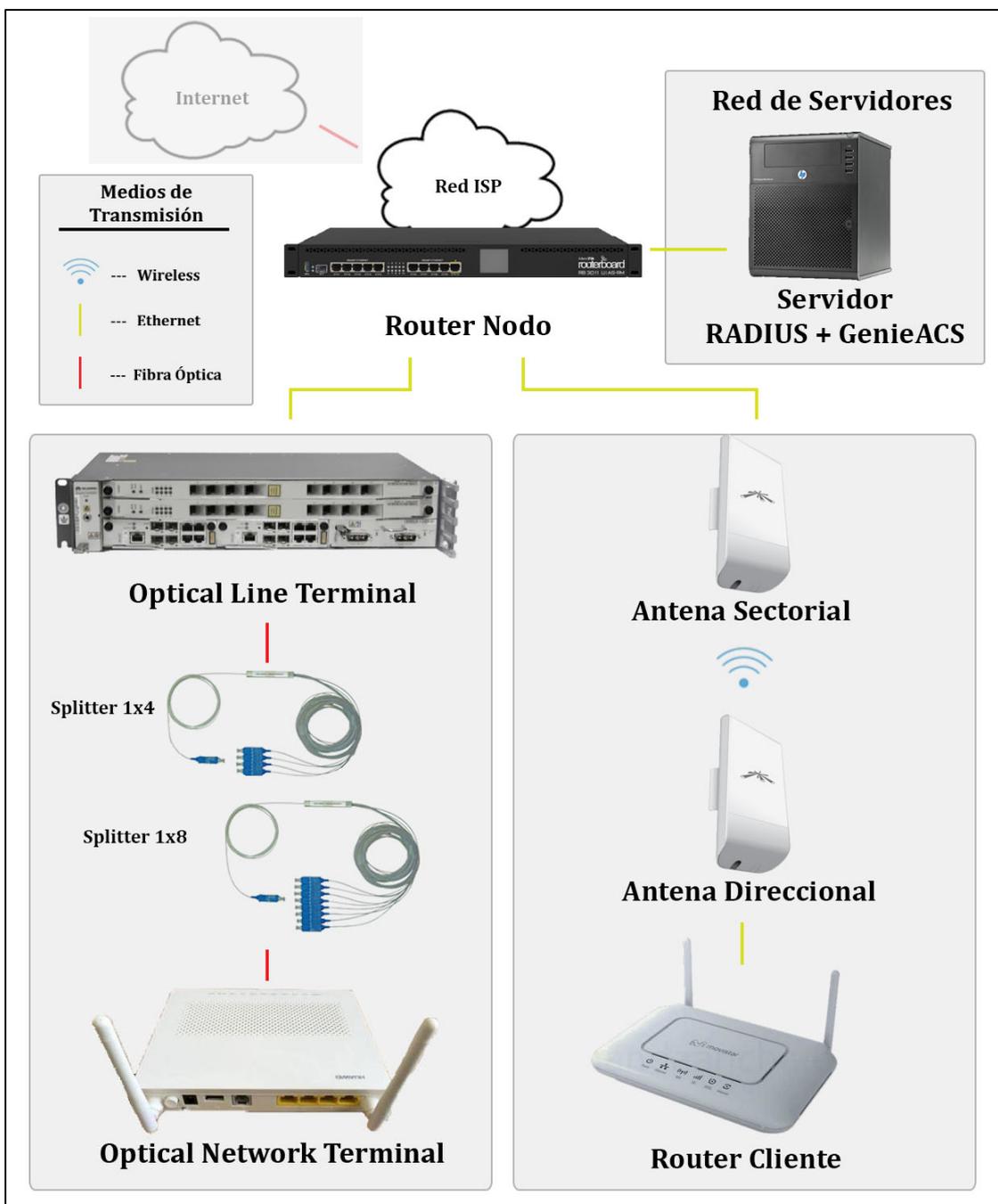


Figura 35 – Dispositivos utilizados en la Red del Laboratorio

La red consiste en: un **Router Nodo** (*Mikrotik RB3011*), en este router se configura filtrado de paquetes (firewalling), calidad de servicio (QoS) y se agrega tráfico de los routers CPE. Aquí también es donde se encuentra el **servidor PPPoE** para que los clientes puedan tener acceso a Internet.

Conectado al **Router Nodo** tenemos 3 dispositivos:

- Un **Optical Line Terminal** (*Huawei SmartAX MA5608T*), el Network Distribution Device para la red FTTH.
- Una **Antena Sectorial** (*Ubiquiti NanoStation Loco M5*), para la red WIMAX.
- Un **Servidor RADIUS + GenieACS** (*HP Proliant Microserver*).

En la red **WIMAX** encontramos nuestra Antena Sectorial, esta antena se conecta con una **Antena Direccional** (*Ubiquiti NanoStation Loco M5*), en nuestro caso utilizamos la misma antena que la Antena Sectorial para el laboratorio. Conectamos la Antena Direccional a un **Router CPE** (*NuCom WR644GV*), el router que el cliente tendrá en su casa.

En la red **FTTH** se encuentra la OLT que se conecta a un **Optical Network Terminal** (*Huawei EchoLife HG8546M*) mediante **Splitters**. En el caso del laboratorio, utilizamos primero un **Splitter 1x4** y a continuación un **Splitter 1x8**, haciendo que por cada puerto GPON de la OLT, podríamos tener 32 posibles clientes. La ONT que utilizamos permite utilizarlo como **Router CPE** también.

Tanto en la red WIMAX como la red FTTH, el router CPE se configurará con unos datos de conexión **PPPoE** que permite acceder al internet y a los servicios ofrecidos por el ISP.

Por último, el **Servidor RADIUS + GenieACS** es donde se encuentra nuestro servicio **ACS** (*GenieACS*), capaz de gestionar, aprovisionar y monitorizar nuestros Routers CPE, y nuestro servicio **RADIUS** (*FreeRadius*), permite gestionar los usuarios y su uso de los servicios ofrecidos por el ISP.

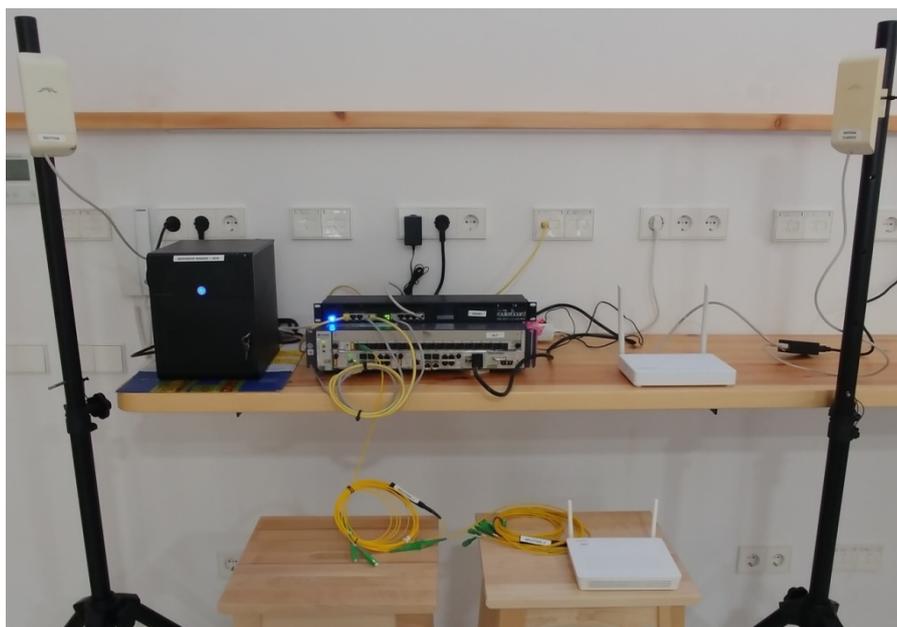


Figura 36 – Montaje de Red del Laboratorio

3.1.2. Direccionamiento IP

El Direccionamiento IP en nuestra red está planteado de la manera siguiente:

La red tiene **servicios** y **nodos** que deben distinguirse unos de otros, se opta por un direccionamiento de tipo “**172.Servicio.Nodo.X**”.

Primero tenemos que definir los **servicios** que vamos a tener en nuestra red. Un ISP podría ofrecer servicios como **Producción** (para acceso a Internet), **VoIP** (para telefonía), **Cámaras** (para cámaras de video vigilancia), etc. pero en nuestro laboratorio solo utilizamos el servicio de **Producción**.

Una vez definido los servicios, empezamos a definir las direcciones IP que tendrá nuestra red. Empezamos por las redes de **Gestión**, estas redes nos permitirán gestionar/configurar los CPE de los clientes remotamente. Especificamos después, un direccionamiento IP para nuestros **Servidores**. Por último, definimos un direccionamiento IP para **Producción**, para el acceso a Internet de los clientes. En caso de que se agoten las IPs Públicas habrá un Pool de IPs Privadas.

RED	Direccionamiento IP	VLAN
Gestión WIMAX	172.16.X.0/24	29
Gestión FTTH	172.17.X.0/24	99
Gestión OLT	172.18.X.0/30	--
TR-069	172.24.X.0/24	--
Red de Servidores	192.168.254.0/24	--
Producción WIMAX	Públicas	10
	172.30.X.0/24	20
Producción FTTH	Públicas	10
	172.31.X.0/24	20

Figura 37 – Tabla Direccionamiento IP Servicios

En cada **Población** tenemos un Router de Cabecera (**nodo**) a la que asignamos un ID propio. Asignamos este **ID** para poder distinguir los nodos más fácilmente (si vemos en un Log un error con la dirección IP 172.30.8.150 sabemos a simple vista que hay un fallo con una IP de Producción Wimax en el nodo en la Población E).

Id Nodo	Población
0	A
2	B
4	C
6	D
8	E
10	F
12	G

Figura 38 – Tabla ID Nodo - Población

Entre cada ID Nodo dejamos un hueco por si crece la red en dicha población.

3.2. Auto-aprovisionamiento

Buscamos un servicio capaz de aprovisionar una configuración preestablecida a los dispositivos finales de los clientes. Este servicio de auto-aprovisionamiento se conoce como **Auto Configuration Server**, como explicamos previamente.

De todas las opciones elegimos **GenieACS** debido a que actualmente todavía se está actualizando y manteniendo, mientras que las otras opciones dejaron de recibir mantenimiento en 2016. Además, GenieACS con la última actualización v1.1 ha implementado scripts para facilitar la creación de **Presets**, ajustes preestablecidos basados en importancia, condiciones previas que se deben cumplir y las acciones que se ejecutarán, que serán aprovisionados a los dispositivos CPE.

Una vez elegida la solución ACS instalamos sobre nuestro servidor, un **HP Proliant Microserver** con el Sistema Operativo **Debian**.

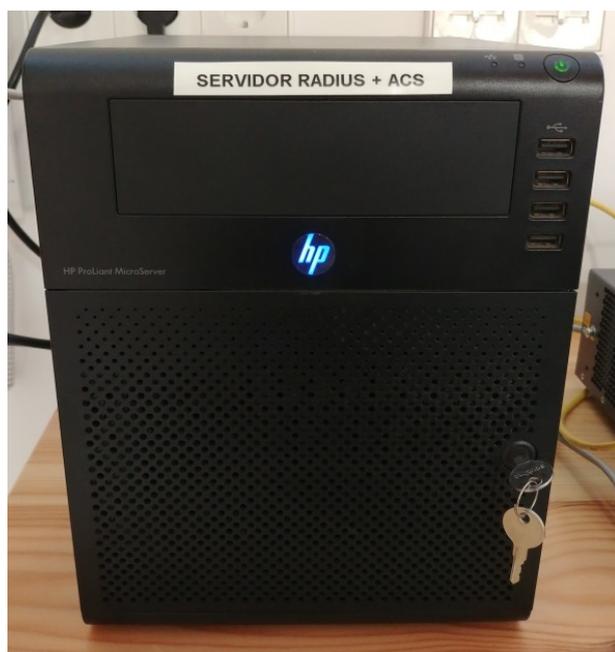


Figura 39 – Servidor HP Proliant Microserver

3.2.1. Servidor GenieACS

GenieACS es una solución de gestión remota TR-069 open-source. Existe una capa **Backend** donde se encuentra todo el código y una capa **Frontend** opcional. En GenieACS la capa **Backend** se refiere como **Core** y la capa **Frontend** como **GUI**.

La página oficial de GenieACS viene con mucha ayuda, **indicando posibles fallos** que puedan salir durante la instalación, software complementario y un foro donde un usuario pueda realizar preguntas sobre GenieACS.

Para la instalación de GenieACS necesitamos:

- Unos **paquetes básicos** para las dependencias que tiene GenieACS para su instalación y unos programas para la edición de ficheros XML.
- **MongoDB**, una base de datos para guardar los datos de los CPE y los presets configurados.
- **Ruby**, un lenguaje de programación orientado a objetos para la compilación de los paquetes de GenieACS.
- **Node.js** para toda la parte de Web de GenieACS.

Con esto ya tenemos GenieACS y podemos empezar a aprovisionar los dispositivos utilizando el terminal del servidor.

```
001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
001EE3-NC%2DWR644GV-1608001876: New device registered
001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 40 – Terminal del Servidor GenieACS

Pero podemos instalar **GenieACS-GUI**, una versión front-end de GenieACS opcional donde vamos a poder ver toda la configuración de manera más visual. La alternativa a esto era utilizar la consola del servidor para modificar la configuración de los CPE, pero resultaba más fácil de entender y configurar con la GUI.

Cuando instalamos GenieACS realmente estamos instalando 3 módulos: GenieACS-cwmp, GenieACS-nbi y GenieACS-fs.

- **GenieACS-CWMP** es el servicio con el que se comunicaran los CPE, su puerto predefinido es el 7547 TCP/UDP.
- **GenieACS-NBI** es el modulo que se utiliza para comunicar con el Front End.
- **GenieACS-FS** es el servidor de archivos desde el que los CPE descargarán imágenes de firmware y otros tipos de archivos.

Cuando esté todo instalado ya podemos acceder al **front-end** introduciendo en un navegador la IP que tiene el servidor accediendo por el puerto 3000 TCP.

Cuando estemos dentro podemos observar que aparece el **estado de los dispositivos** que hemos aprovisionado. Esta es la página por defecto, **sin vista de administrador**.

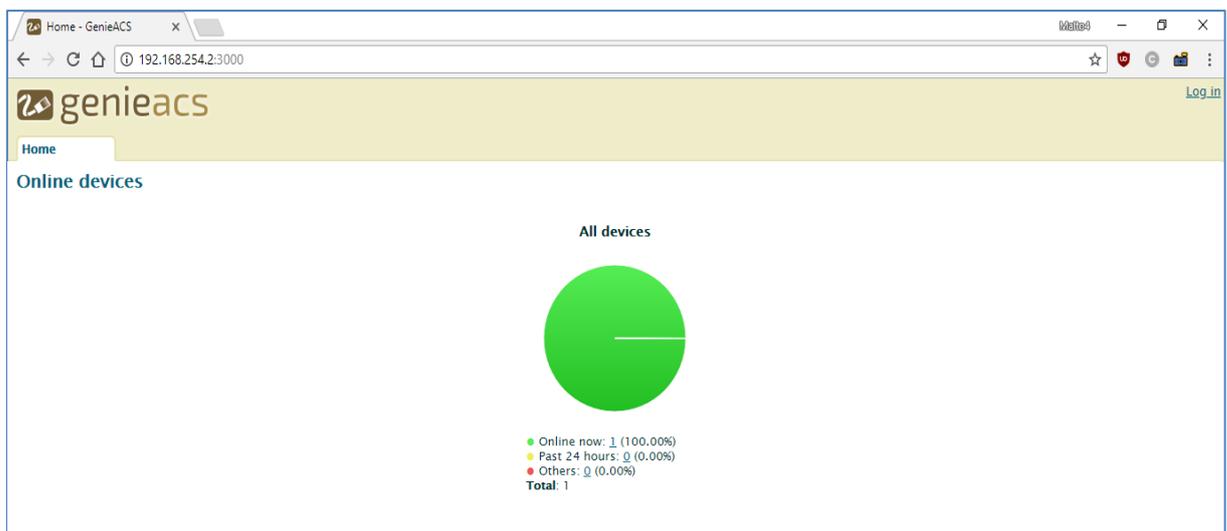


Figura 41 – GenieACS-GUI – Vista sin Usuario

Para poder observar más información tenemos que acceder como **administrador** pulsando en el botón “Log in”.

Una vez hayamos entrado como admin encontramos las pestañas siguientes:

- **Devices** (lista de los CPE que han realizado una conexión con el ACS)
- **Faults** (lista de fallos que han ocurrido durante los aprovisionamientos)
- **Presets** (Ajustes preestablecidos que se pueden aprovisionar a los CPE)
- **Objects** (Objetos que podemos aprovisionar a un CPE)
- **Files** (Actualizaciones Software o Firmware que podemos aprovisionar a un CPE).

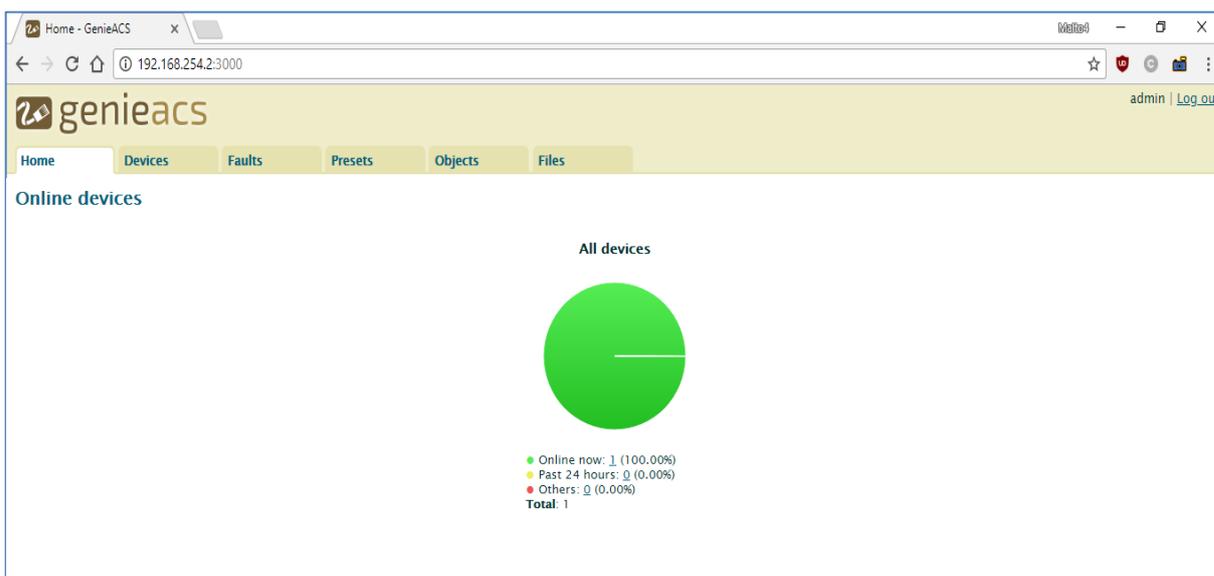


Figura 42 – GenieACS-GUI - Vista Administrador

Devices

La pestaña **Devices** aporta una visión general de todos los CPEs que hayan conectado al menos una vez al ACS. En esta pestaña aparecen todos los dispositivos, tanto los dispositivos que estén online como los que hayan perdido la conectividad con GenieACS. En esta pestaña se pueden aplicar filtros para la búsqueda de algún CPE en concreto.

En nuestra versión de laboratorio ha sido necesario modificar esta página, añadiendo los campos **MODELO** (el modelo de CPE) y **ACTUALIZAR** (el intervalo de tiempo entre cada Inform) para mostrar información más relevante a la empresa.

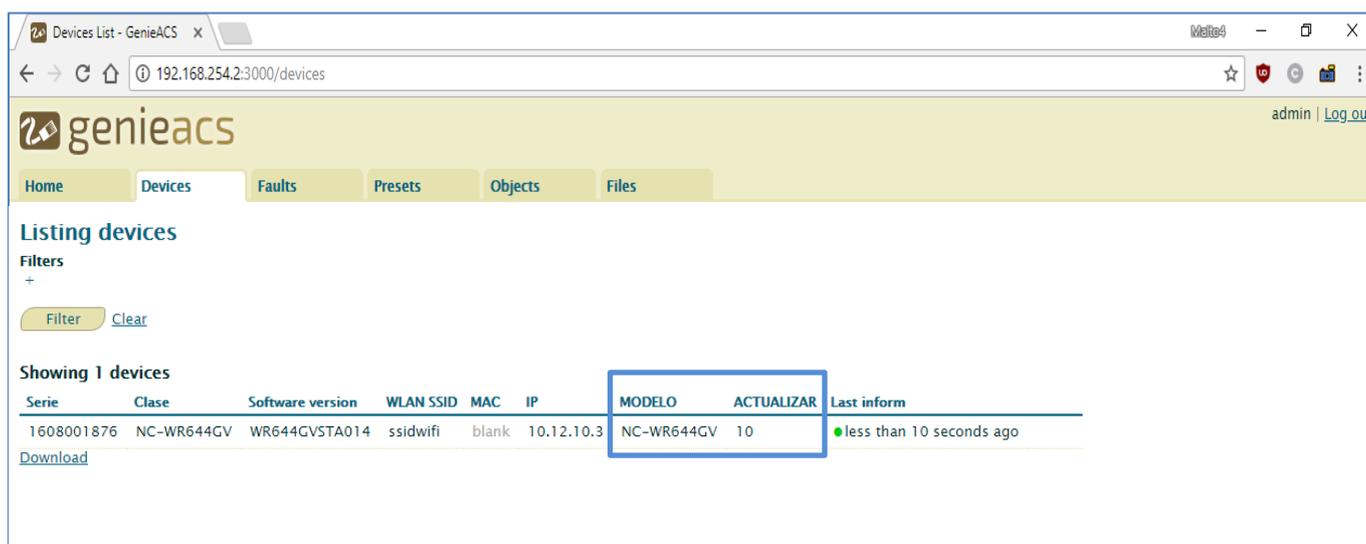


Figura 43 - Pestaña Devices

Si se hace clic sobre un CPE listado aparece una página con información detallada sobre dicho CPE.

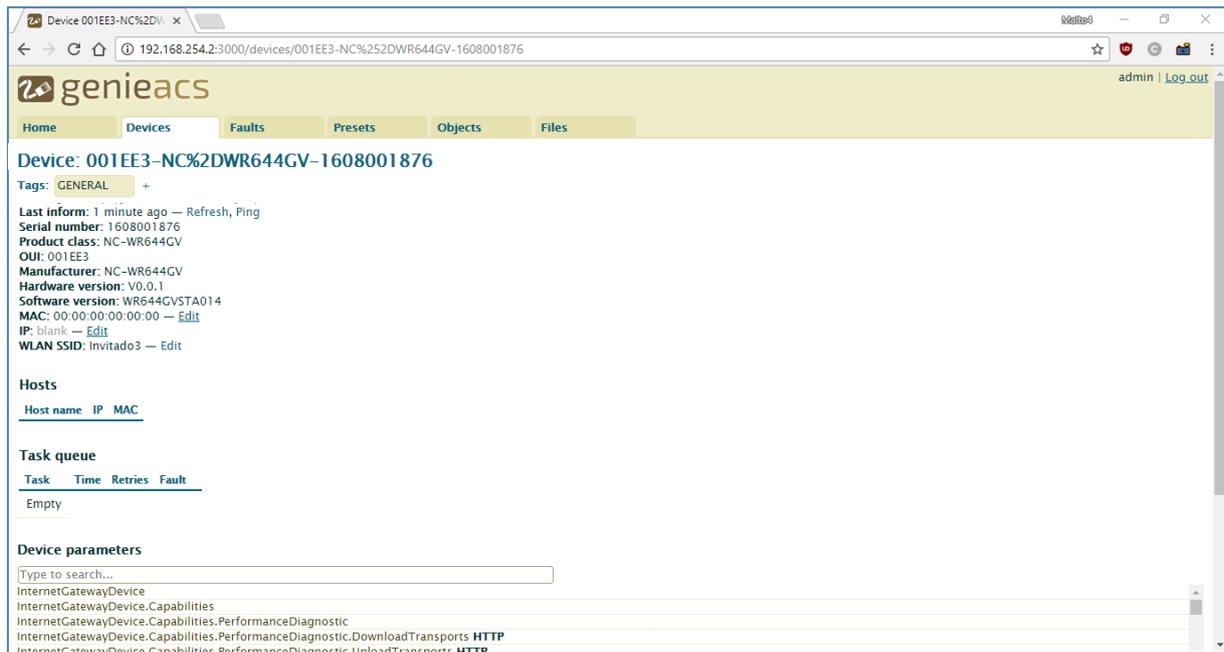


Figura 44 - Pestaña Devices – Información sobre un CPE

Aquí dentro podemos ver al principio los **Tags**, unos **identificadores** que se puede utilizar con el fin de agrupar dispositivos para **filtrar** una búsqueda y especificar **condiciones** para un Preset, que tiene asignado el dispositivo CPE y un resumen de los parámetros más importantes del CPE.

Los **Tags** no son una funcionalidad de TR-069, sino un mecanismo implementado por GenieACS.



Figura 45 - Pestaña Devices – Tags CPE

A continuación, observamos algunos parámetros que definen el CPE. Estos parámetros contienen información específica del CPE:

- **Last inform** (tiempo desde que el CPE realizó su último Inform)
- **Serial number** (numero Serial del CPE)
- **Product class** (modelo del CPE)
- **OUI** (número Organizationally unique identifier del CPE)
- **Manufacturer** (fabricante del CPE)
- **Hardware version** (versión de firmware del CPE)
- **Software version** (la versión de software del CPE)



Figura 46 - Pestaña Devices – Parametros Summary CPE

Además de estos parámetros, añadimos **MAC** (dirección MAC del CPE), **IP** (dirección IP del CPE) y **WLAN SSID** (nombre asignado a la red wireless).

Por último, aparecen tres tablas: la tabla de Hosts, Task Queue y Device Parameters.

- La tabla **Host** muestra los usuarios que están conectados a dicho CPE, en caso de ser un router aquí podríamos ver que dispositivos están conectados a dicho router y con qué direcciones IP.
- La tabla **Task Queue** se utiliza para mostrar las tareas que tiene pendiente el CPE. En caso de realizar pruebas para ver por qué no funciona el aprovisionamiento al CPE aquí podríamos ver en qué punto deja de funcionar.

- La tabla **Device Parameters** se utiliza para observar todos los parámetros especificados en el documento TR-098 y los valores que tienen asignados en el CPE. En esta última tabla podemos filtrar los parámetros que queremos observar.



Figura 47 - Pestaña Devices – Tablas Host, Task Queue y Device Parameters

A final de esta página aparecen algunos botones que realizan funciones RPC:

- **Reboot** (reinicia el CPE)
- **Factory Reset** (realiza un reset de fábrica)
- **Push File** (permite enviar un fichero al CPE)
- **Add Firmware** (permite enviar una Imagen de Firmware al CPE)
- **Delete** (elimina el CPE de la base de datos de GenieACS).

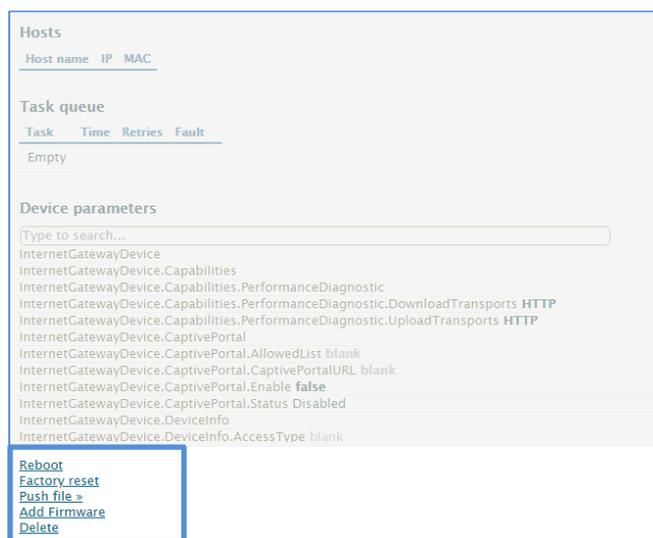


Figura 48 - Pestaña Devices – Botones RPC

Faults

La pestaña **Faults** tiene el objetivo de mostrar una lista completa de todos los fallos actuales (**tareas defectuosas**) reconocidas por GenieACS. Los fallos que aparecen en la tabla Task Queue de cada CPE también aparecen aquí. En caso de un problema importante, esta lista podría obtener muchas entradas en un corto periodo de tiempo. Para eso, en caso de querer **buscar** un fallo en concreto, podemos, con el uso del filtro, especificar una búsqueda por Dispositivo, Tarea y Fecha.

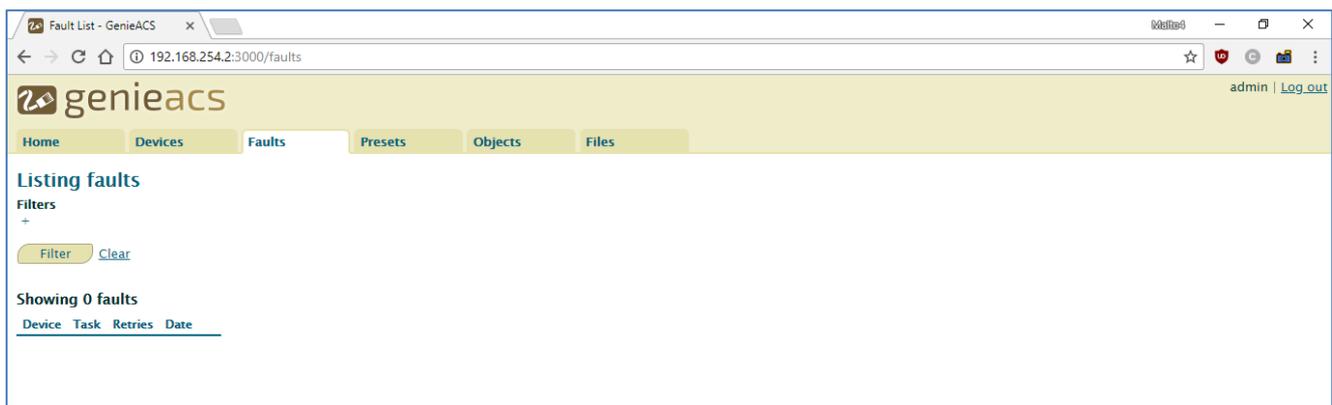


Figura 49 – Pestaña Faults

Presets

La pestaña **Presets** permite al técnico crear **ajustes preestablecidos** basados en importancia (**peso**), condiciones previas que se deben cumplir (**precondiciones**) y las acciones que se ejecutaran (**configuraciones**).

Cuando creas un nuevo Preset aparece la página siguiente:

Figura 50 – Pestaña New Preset

Aquí podemos especificar un nombre a nuestro Preset y el peso que debería tener.

A continuación, podemos especificar **precondiciones**, los prerequisites que un CPE tiene que cumplir para formar parte de este Preset. Algunos de los parámetros que podemos especificar como prerequisites son los siguientes.

Precondition

+

- Last inform
- Tag
- Serial number
- Product class
- OUI
- Manufacturer
- Hardware version
- Software version

Precondition

Tag = x

OUI = x

+

Figura 51 – Pestaña Preset - Preconditions

Por último, podemos elegir qué **configuración** deberíamos aprovisionar a los CPE que cumplan este Preset.

- **Set** se utiliza para asignar valores a parámetros definidos en el documento TR-098.
- **Refresh** es para actualizar el valor de un parámetro cada X tiempo (definido por el técnico).
- **Add/Remove Tag** se utiliza para añadir o eliminar un identificador a los CPE que ejecutan el Preset.
- **Add/Remove Object** se utiliza para añadir o eliminar instancias de objetos en el CPE.
- **Command value/age** se utiliza para ejecutar un comando, especificando el valor que debería tener o cada cuanto se debería ejecutar.
- **Software versión** es para actualizar la versión de software del CPE.

Configurations

+

- Set
- Refresh
- Add tag
- Remove tag
- Add object
- Remove object
- Command value
- Command age
- Software version

Configurations

Add object to x

Add tag x

Remove tag x

+

Figura 52 – Pestaña Preset - Configurations

Objects

La pestaña **Objects** se utiliza para poder crear instancias de objetos, un contenedor para parámetros y/u otros objetos.

El **nombre completo de la ruta** de un **parámetro** viene dado por el **nombre del parámetro**, adjunto al **nombre completo de la ruta del objeto** en el que está contenido. Para diferenciar las distintas **instancias** de objetos, cada objeto tiene un número para identificarse (InternetGatewayDevice.WANDevice.2, InternetGatewayDevice.WANDevice.3, etc.)

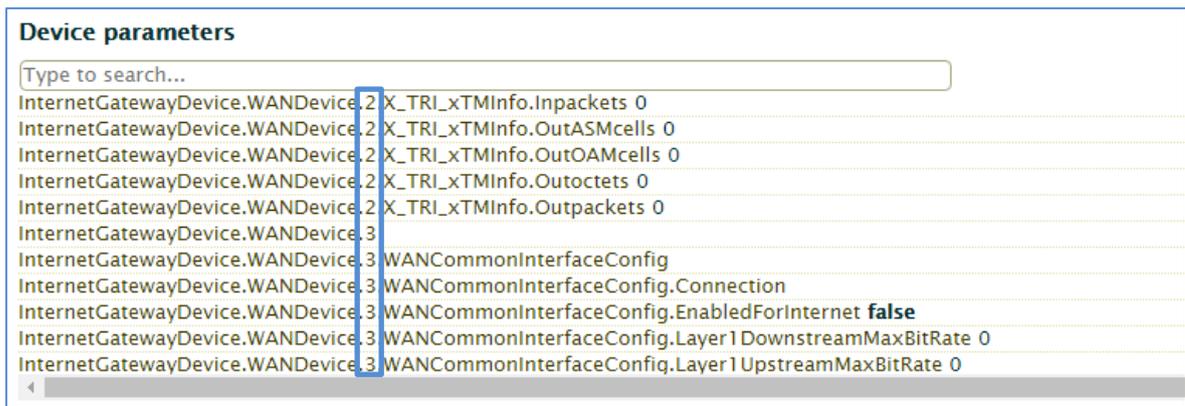


Figura 53 – Tabla Device Parameters

En la pestaña **Objects** nos aparece una lista de todos los objetos creados. Aquí también podemos crear un nuevo objeto pulsando en el botón “New Object”. Un ejemplo de un objeto sería:



Figura 54 – Pestaña Objects – PPPObject Object

Aquí dentro podemos especificar un **nombre** para nuestro objeto y empezar a **asignar valores a parámetros**.

Los atributos que nos aparecen son: Key, Parameter y Value.

- **Key** se utiliza para especificar si vamos a crear un objeto nuevo o modificar uno ya existente. Utilizamos el atributo Key para buscar una instancia de objeto existente y modificarla si es necesario para que coincida con las definiciones de objeto.
- **Parameter** se utiliza para especificar el parámetro al que queremos asignar/modificar un valor.
- **Value** se utiliza para especificar un valor a dicho parámetro.

Files

La pestaña Files se utiliza para guardar todos los archivos de configuración que se podría aportar a un CPE. Se pueden guardar archivos de configuración de tipo:

- **Actualización de Firmware** (para actualizar el firmware del CPE)
- **Contenido Web** (para el Portal de cliente)
- **Vendor Configuration File** (para sobrescribir toda la configuración del CPE)
- **Archivo de tono** (para teléfonos VoIP).
- **Archivo de timbre** (para teléfonos VoIP).

Al entrar en la pestaña Files nos aparece una lista de todos los archivos de configuración que estén subidos. Pulsando en el botón “New File” podemos subir un nuevo archivo de configuración.

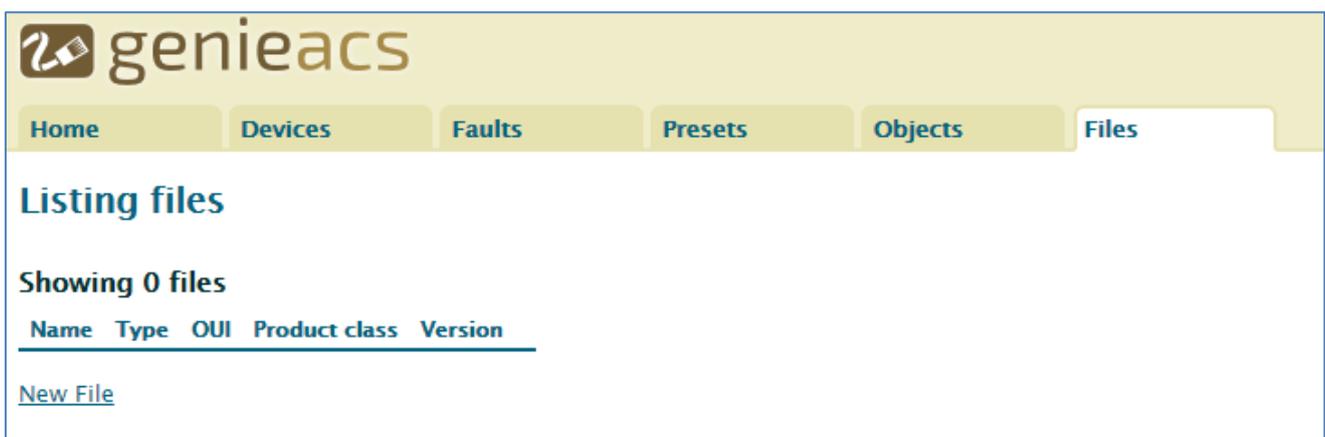


Figura 55 – Pestaña Files



The screenshot shows the 'genieacs' web interface. At the top, there is a navigation bar with tabs: Home, Devices, Faults, Presets, Objects, and Files. The 'Files' tab is active. Below the navigation bar, the page title is 'New file'. The form contains the following fields:

- File type: 1 Firmware Upgrade Image (dropdown menu)
- OUI: E48D8C (text input)
- Product class: hAP ac lite (text input)
- Version: 6.38.1 (text input)
- File: Choose File (button) No file chosen (text)

At the bottom of the form, there are two buttons: 'Upload' (green) and 'Back' (blue).

Figura 56 – Pestaña Files – New File

Al crear un nuevo archivo de configuración nos aparecen los campos siguientes:

- **File type** es el tipo de archivo que vamos a subir, este puede ser uno de los 5 tipos previamente dichos.
- **OUI** para especificar la marca del CPE.
- **Product Class** para especificar el modelo del CPE.
- **Versión de software** que tiene el CPE.
- **Archivo de configuración** que vamos a subir para aprovisionar a los CPE que tengan el OUI, Product Class y Version previamente dichos.

3.2.2. Presets Configurados

Empezamos a configurar los presets, primero diseñando los pasos por los que deberían pasar los CPE.

El **primer paso** debería ser alguna manera de poder hacer que todas las comunicaciones posteriores vayan **encriptados**, de manera que nadie pueda meterse por medio y escuchar los mensajes. Este primer paso debería ser **genérico** para que se pueda aplicar a todos los CPE independiente del fabricante.

El **segundo paso** debería servir para una configuración **general de cada fabricante** de router. De esta manera podemos tener que todos los Routers Mikrotik o que todos los Routers NuCom tengan una configuración base igual.

El **tercer paso** ya sería configuración **específica** para cada **cliente**. En este tercer paso pondríamos los **datos de conexión** de los clientes y datos específicos del cliente.

El **cuarto paso** sería para iniciar la conexión PPPoE del cliente, dándole acceso a Internet.

Con este diseño un CPE pasaría por diferentes etapas, cada etapa añadiendo configuración a la anterior, haciendo así, de manera más visual, los fallos en el aprovisionamiento para poder ver en qué etapa se haya quedado.

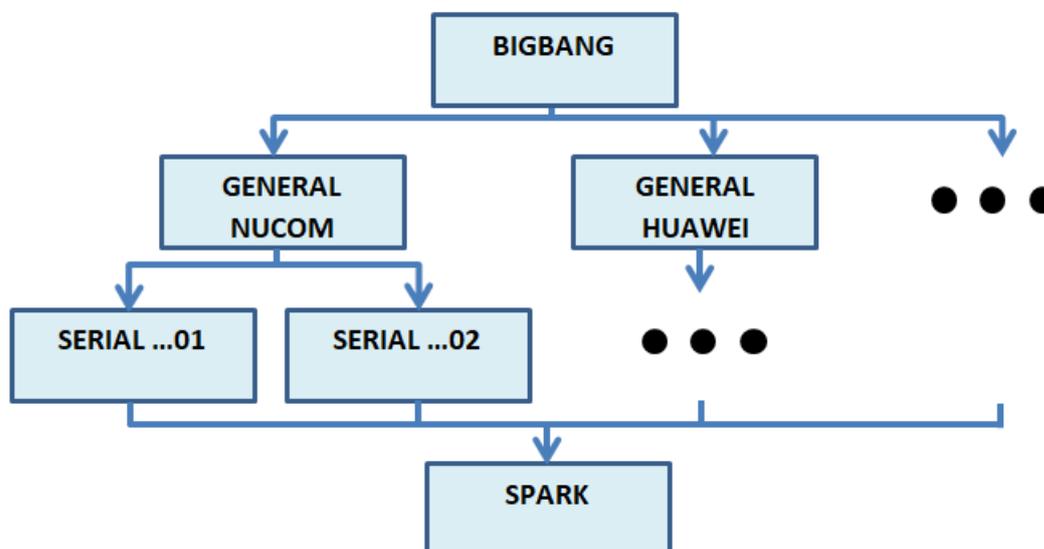


Figura 57 - Gráfica Presets

Realizando esta configuración obtenemos nuestros 4 presets:

- **BigBang:** El router realiza su primera conexión con el servidor GenieACS y empieza el proceso de aprovisionamiento de manera segura. La conexión segura se realizó mediante el uso de un parámetro de TR-069: **ConnectionRequest**. Con este parámetro si el username y password que le asignamos tanto al CPE como a GenieACS no coinciden no se realiza la conexión, el CPE lo rechaza. En este paso también asignamos un **Tag** para especificar en que Preset estamos.
- **General:** Empieza a crear los objetos necesarios para cada marca de Router. Para este Preset General tenemos que tener un **Objeto** definido previamente. Quitamos los Tags previos y especificamos el Tag actual (GENERAL).
- **Serial:** Preset basándonos en el número serial de los CPE para aprovisionar una configuración específica de cliente. Cada cliente tendrá su Preset Serial con sus propios datos de **conexión a Internet con PPPoE** y, opcionalmente, la SSID que va a tener CPE y su contraseña. Por último, se añade un Tag especificando el Preset (SERIAL).
- **Spark:** Este Preset es genérica para todas las marcas de Router. Sirve para **empezar los servicios (PPPoE)** configurados en los Presets anteriores.

Fases

Cualquier dispositivo CPE con estos Presets pasaría por las siguientes fases:

- 1- El dispositivo CPE empieza una conexión con GenieACS. Como es la **primera comunicación** con el servidor, el CPE envía un “**BOOT**” Inform.

```
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 58 – Fase 1 – BOOT Inform

- 2- GenieACS registra este **dispositivo nuevo** y empieza a enviar la función RPC “**RefreshObject**” para obtener la configuración actual del dispositivo.

```
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 59 – Fase 2 – Dispositivo Registrado

- Una vez acabada la función RPC “RefreshObject”, el servidor nota que el dispositivo cumple con los prerequisites para el primer **Preset (BIGBANG)** y aporta al dispositivo la configuración especificada mediante el uso del RPC “setParameterValues”.

```
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 60 – Fase 3 – Preset BIGBANG

- El dispositivo CPE con esta configuración empieza a enviar “PERIODIC” Informes al servidor. Estos Informes se envían al servidor para informar de que el dispositivo CPE sigue en línea.

```
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 61 – Fase 4 – PERIODIC Inform

- 5- El servidor nota que el dispositivo cumple los prerequisites del **Preset GENERAL** y le aporta la configuración especificada en dicho Preset. En el Preset GENERAL el servidor aporta al dispositivo CPE un **Objeto** para PPPoE mediante el RPC **“addObject”**.

```

25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0

```

Figura 62 – Fase 5 – Preset GENERAL

- 6- El dispositivo CPE vuelve a mandar **“PERIODIC”** Informos una vez recibida la configuración del Preset GENERAL para informar que sigue activo.

```

25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0

```

Figura 63 – Fase 6 – PERIODIC Inform

- 7- GenieACS aporta la configuración del **Preset SERIAL** específico para el dispositivo CPE. Dependiendo del número de serie del dispositivo se aprovisionará una configuración de un Preset SERIAL u otra.

```
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 64 – Fase 7 – Preset SERIAL

- 8- El dispositivo CPE una vez recibida la configuración del Preset SERIAL realiza un **“VALUE CHANGE”** Inform, informando al servidor que un valor ha cambiado. Esto es debido a la configuración de **PPPoE** que le aportamos al dispositivo. A continuación el dispositivo CPE envía **“PERIODIC”** Inform al servidor.

```
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 65 – Fase 8 – VALUE CHANGE Inform

- 9- El servidor nota que ya tiene la configuración del Preset SERIAL y envía la configuración del Preset SPARK, levantando la conexión PPPoE y dando al dispositivo CPE **acceso a Internet**. El dispositivo CPE continúa enviando **“PERIODIC”** Informs al servidor para informar que sigue funcionando y está en línea.

```
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Inform (1 BOOT); retry count 0
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: New device registered
25 Jan 13:29:49 - 001EE3-NC%2DWR644GV-1608001876: Started task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:20 - 001EE3-NC%2DWR644GV-1608001876: Completed task refreshObject(5a69cddb80cc19cb097e4080)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(BIGBANG) -()
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:22 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce5680cc19cb097e4081)
25 Jan 13:32:35 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(GENERAL) -(BIGBANG)
25 Jan 13:32:45 - 001EE3-NC%2DWR644GV-1608001876: Started task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task addObject(5a69ce6d60dbd8ca090b38f0)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:11 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce6d60dbd8ca090b38f1)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(SERIAL) -()
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:33 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69ce9d37a95bc709d31054)
25 Jan 13:33:50 - 001EE3-NC%2DWR644GV-1608001876: Inform (4 VALUE CHANGE); retry count 0
25 Jan 13:35:31 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Presets discrepancy found
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Updating tags +(KINDLING) -(SERIAL,GENERAL)
25 Jan 13:37:11 - 001EE3-NC%2DWR644GV-1608001876: Started task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task setParameterValues(5a69cf7760dbd8ca090b38f2)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Started task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:37:12 - 001EE3-NC%2DWR644GV-1608001876: Completed task getParameterValues(5a69cf7760dbd8ca090b38f3)
25 Jan 13:38:53 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
25 Jan 13:40:33 - 001EE3-NC%2DWR644GV-1608001876: Inform (2 PERIODIC); retry count 0
```

Figura 66 – Fase 9 – VALUE CHANGE Inform

3.2.3. TFTP

Para poder aprovisionar la **dirección IP del servidor ACS** al dispositivo CPE para **redes WIMAX/Ethernet**, se estudiaron las diferentes posibilidades.

Encontramos la **Opción 66**, una Opción DHCP que aprovisiona a los dispositivos CPE la dirección IP de donde se encuentra un **servidor TFTP**, servidor que permite a un cliente descargar o subir un fichero de tamaño pequeño mediante el **protocolo TFTP**.

Utilizamos este servidor TFTP para aprovisionar ficheros de configuración con la **dirección IP del servidor ACS** a los CPE. Configuramos el servidor TFTP en nuestro servidor RADIUS + GenieACS para guardar dichos ficheros de configuración.

De esta manera el CPE cuando se conecte a la red recibirá su **dirección IP** por **DHCP** y la **dirección IP del servidor TFTP** por **Opción 66 DHCP**. Una vez tenga la dirección IP del servidor TFTP busca un fichero de configuración compatible con el CPE y **descarga la configuración** (la **dirección IP del servidor ACS**). Por último, el CPE ya dispone de la dirección IP del servidor ACS y empieza una conversación con el servidor ACS.

3.2.4. Página Web

Realizamos cambios a **GenieACS-GUI** para poder visualizar unos campos adicionales en la página Devices que resultaban más importantes a la empresa.

Como explicamos previamente, en la página Devices añadimos los campos:

- **MODELO** (Modelo del dispositivo CPE)
- **ACTUALIZAR** (Tiempo que espera entre cada Inform – PeriodicInformInterval).



Figura 67 - Pestaña Devices Original y Modificada

Además de los campos en la vista detallada de un CPE:

- **MAC** (dirección MAC del CPE)
- **IP** (dirección IP del puerto WAN)
- **WLAN SSID** (nombre asignado a la red wireless).

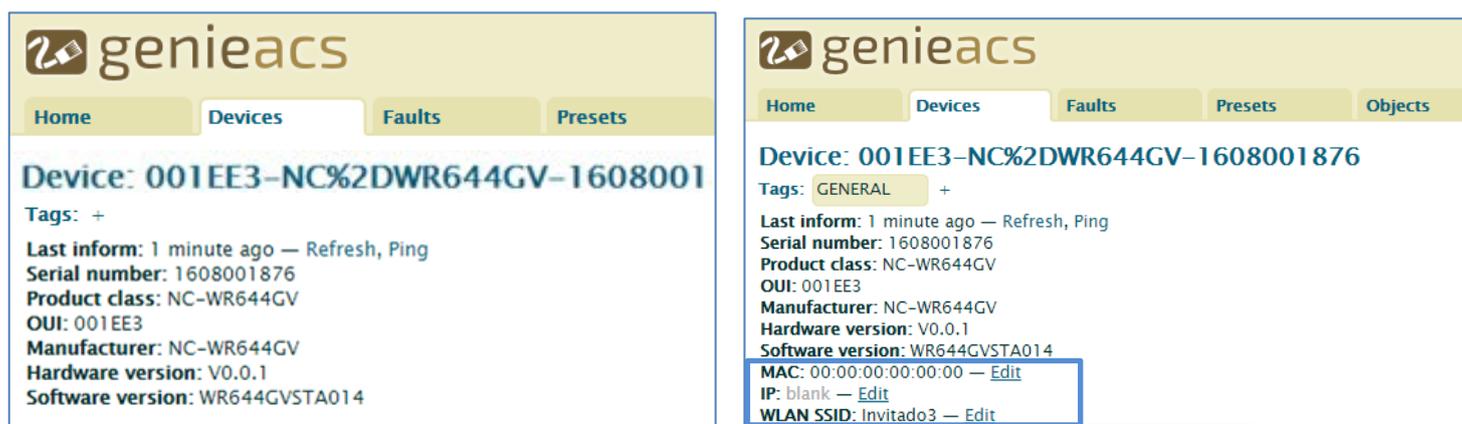


Figura 68 - Pestaña Devices Original y Modificada– Información CPE

Para realizar dichos cambios, tuvimos que modificar algunos ficheros de configuración (**yml**), ficheros utilizados para modificar ciertos parámetros visuales,

de GenieACS-GUI. Para añadir dichos campos a la Página Devices modificamos el fichero **index_parameters.yml**, añadiendo lo siguiente:

```
Serie: InternetGatewayDevice.DeviceInfo.SerialNumber
Clase: InternetGatewayDevice.DeviceInfo.ProductClass
Software version: InternetGatewayDevice.DeviceInfo.SoftwareVersion
WLAN SSID: InternetGatewayDevice.LANDevice.1.WLANConfiguration.1.SSID
MAC: InternetGatewayDevice.DeviceInfo.MACAddress
IP: InternetGatewayDevice.WANDevice.2.WANConnectionDevice.1.WANIPConnection.1.ExternalIPAddress
MODELO: InternetGatewayDevice.DeviceInfo.Manufacturer
ACTUALIZAR: InternetGatewayDevice.ManagementServer.PeriodicInformInterval
~
~
~
"index_parameters.yml" 9 lines, 517 characters
```

Figura 69 – index_parameters.yml

Los cambios realizados en la página Devices detallada de un dispositivo CPE se realiza en el fichero **summary_parameters.yml**.

```
Serial number: summary.serialNumber
Product class: summary.productClass
OUI: summary.oui
Manufacturer: summary.manufacturer
Hardware version: summary.hardwareVersion
Software version: summary.softwareVersion
MAC: summary.mac
IP: summary.ip
WLAN SSID: summary.wlanSsid
Hosts:
  _object: InternetGatewayDevice.LANDevice.1.Hosts.Host
  Host name: HostName
  IP: IPAddress
  MAC: MACAddress
~
~
"summary_parameters.yml" 15 lines, 435 characters
```

Figura 70 – summary_parameters.yml

Primero se especifica un **nombre para el campo** que vamos a mostrar en la página. A continuación especificamos el campo correspondiente al dato que queremos mostrar de la **documentación TR-098**.

3.3. Aprovisionamiento de CPEs

En nuestras **pruebas** de laboratorio realizamos pruebas con las tecnologías de red **WIMAX y FTTH** para el aprovisionamiento de los CPEs.

En el caso de **WIMAX** el aprovisionamiento de la **dirección IP del servidor ACS** se realiza con el **servidor TFTP (Opción 66)**. Mientras que en el caso de **FTTH** se realiza con el protocolo propio **OMCI**.

En ambos casos el **CPE recibe** la dirección IP del **servidor ACS** y empieza a comunicar. Pasa por los **cuatro presets configurados**, cada una añadiendo configuración al preset anterior, hasta que el **CPE tenga acceso a Internet** con los datos de conexión específico de cada cliente.

3.3.1. Redes Ethernet/Wimax

En las redes Ethernet y Wimax un dispositivo CPE pasa por los siguientes pasos de configuración:

- 1- El **dispositivo CPE** se conecta a la red ISP y pide direccionamiento IP mediante **DHCP**. El Router Nodo reparte una dirección IP al dispositivo CPE y también la **Opción 66 DHCP**. Con esta Opción 66, informa al dispositivo de la dirección IP del **servidor TFTP**. El dispositivo busca y descarga el **fichero de configuración** del servidor TFTP. Las figuras siguientes demuestran la configuración que tiene el **CPE inicialmente** comparada con la configuración que recibe el CPE del **servidor TFTP**.

NuCom NC-WR644GV Home Gateway

TR069 Settings

Enable TR069:

ACS URL:

ACS User Name:

ACS Password:

Period Inform:

Inform Interval:

Connection Request:

Connection Request Port:

Connection Request URL:

User Name:

Password:

Apply Refresh

Figura 71 - NuCom Configuración TR-069 inicial

NuCom NC-WR644GV Home Gateway

TR069 Settings

Enable TR069:

ACS URL:

ACS User Name:

ACS Password:

Period Inform:

Inform Interval:

Connection Request:

Connection Request Port:

Connection Request URL:

User Name:

Password:

Apply Refresh

Figura 72 -NuCom Configuración TR-069 aportado por TFTP

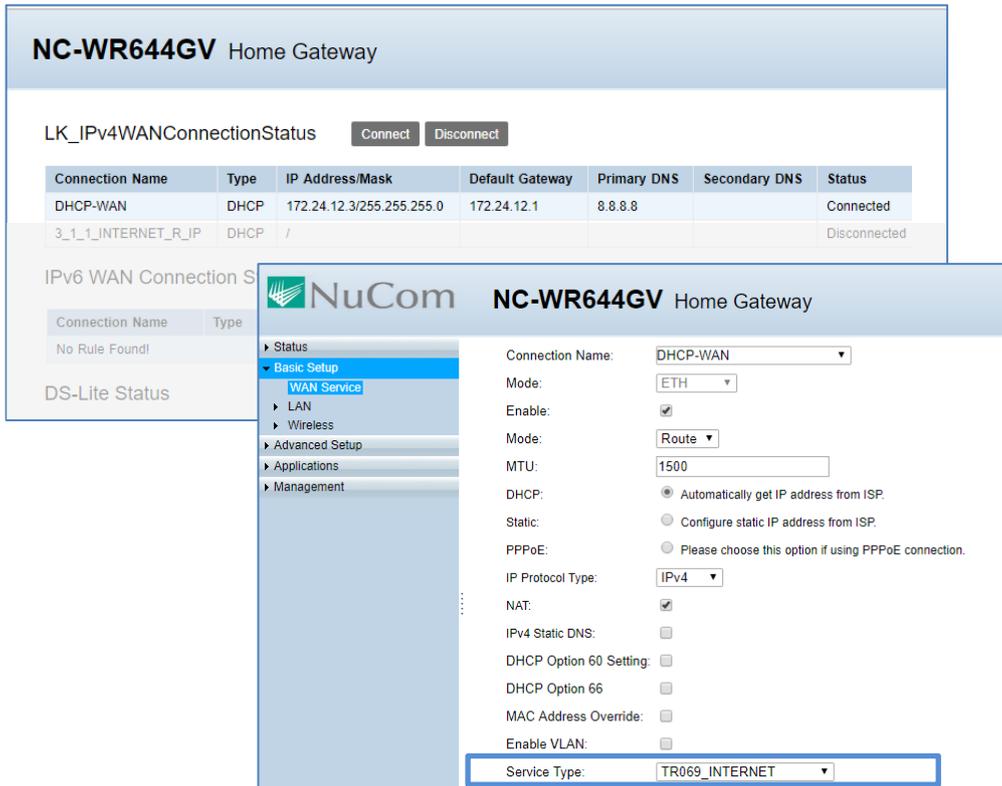


Figura 73 – NuCom Configuración DHCP-WAN por defecto

- Una vez el dispositivo CPE tenga el archivo de configuración provisionado por el servidor TFTP empieza a **comunicar con el servidor ACS**. El servidor ACS **registra** el CPE en su **base de datos** y comprueba las precondiciones de los presets configurados. Al ser la primera comunicación con el servidor ACS el dispositivo CPE cumple el Preset **“BIGBANG”**, preset genérico para todos los dispositivos CPE. Con el **Preset BIGBANG**, el CPE recibe un usuario y contraseña para comunicarse con el servidor ACS de **manera segura**. Además, se añade un **TAG “BIGBANG”** al dispositivo CPE en la base de datos de GenieACS.

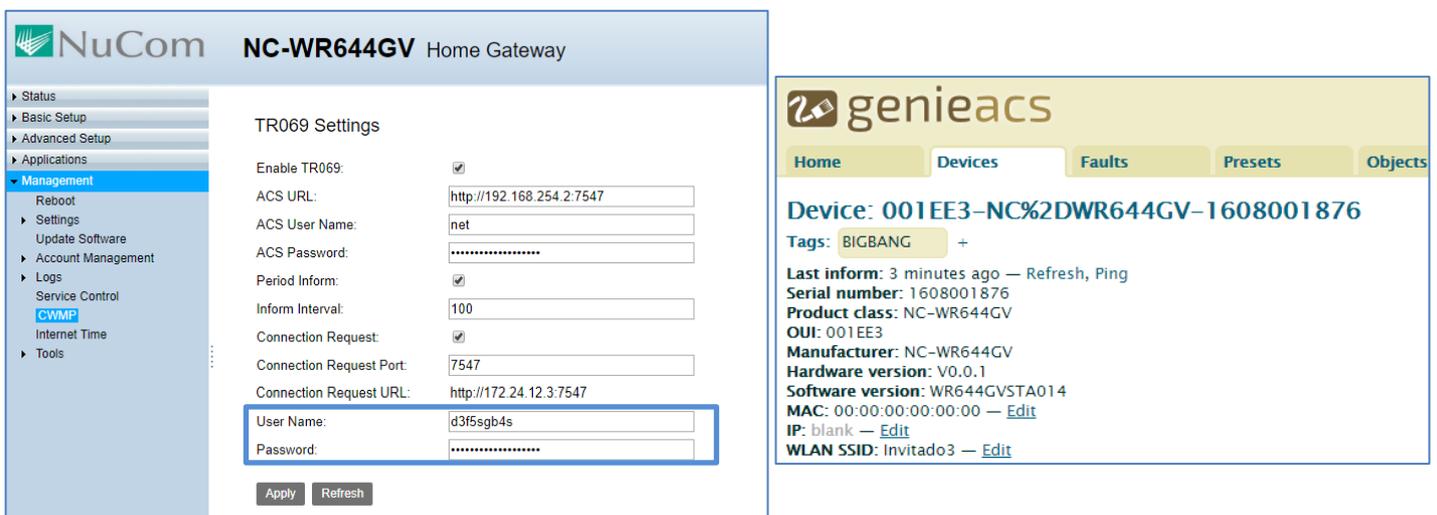


Figura 74 – NuCom Configuración Preset BIGBANG

- 3- A continuación, el dispositivo CPE recibe la configuración del **Preset GENERAL**, un preset **genérico para cada fabricante**. En este Preset GENERAL el dispositivo CPE recibe un **objeto PPP** que contiene un conjunto de parámetros genéricos para **conexiones PPP**. Se le elimina el TAG de “BIGBANG” y le añadimos el TAG “GENERAL”.

NC-WR644GV Home Gateway

LK_IPv4WANConnectionStatus Connect Disconnect

Connection Name	Type	IP Address/Mask	Default Gateway	Primary DNS	Secondary DNS	Status
DHCP-WAN	DHCP	172.24.12.3/255.255.255.0	172.24.12.1	8.8.8.8		Connected
GESTFY	PPP	/				Disconnected
3_1_1_INTERNET_R_IP	DHCP	/				Disconnected

NuCom NC-WR644GV Home Gateway

Basic Setup
WAN Service

Connection Name: **GESTFY**

Mode: **ETH**

Enable:

Mode: **Route**

MTU: **1492**

DHCP: Automatically get IP address from ISP.
 Configure static IP address from ISP.
 Please choose this option if using PPPoE connection.

Static:

PPPoE:

IP Protocol Type: **IPv4**

NAT:

IPv4 Static DNS:

DHCP Option 60 Setting:

DHCP Option 66:

MAC Address Override:

Enable VLAN:

Service Type: **INTERNET**

Bound Ports: LAN1 LAN2 LAN3 LAN4
 SSID1 SSID2 SSID3 SSID4

A grey out port means it has been bound.

genieacs

Home **Devices** Faults Presets Objects

Device: 001EE3-NC%2DWR644GV-1608001876

Tags: **GENERAL** +

Last inform: ● less than 20 seconds ago — Refresh, Ping

Serial number: 1608001876

Product class: NC-WR644GV

OUI: 001EE3

Manufacturer: NC-WR644GV

Hardware version: V0.0.1

Software version: WR644GVSTA014

MAC: 00:00:00:00:00:00 — [Edit](#)

IP: blank — [Edit](#)

WLAN SSID: Invitado3 — [Edit](#)

Figura 75 – NuCom Configuración Preset GENERAL

- 4- Una vez recibida la configuración del Preset GENERAL, el dispositivo CPE cumple los prerequisites para el **Preset SERIAL**. El servidor ACS envía la configuración específica del cliente para su conexión a Internet con **PPPoE** y le añade el TAG “SERIAL”. Además, en este Preset se configura la **conexión DHCP** para que se utilice solamente para conexiones TR-069 y no para Internet.

genieacs

Home **Devices** Faults Presets Objects

Device: 001EE3-NC%2DWR644GV-1608001876

Tags: **GENERAL** **SERIAL** +

Last inform: ● half a minute ago — Refresh, Ping

Serial number: 1608001876

Product class: NC-WR644GV

OUI: 001EE3

Manufacturer: NC-WR644GV

Hardware version: V0.0.1

Software version: WR644GVSTA014

MAC: 00:00:00:00:00:00 — [Edit](#)

IP: blank — [Edit](#)

WLAN SSID: Invitado3 — [Edit](#)

Figura 76 – NuCom Configuración Preset SERIAL – TAG “SERIAL”

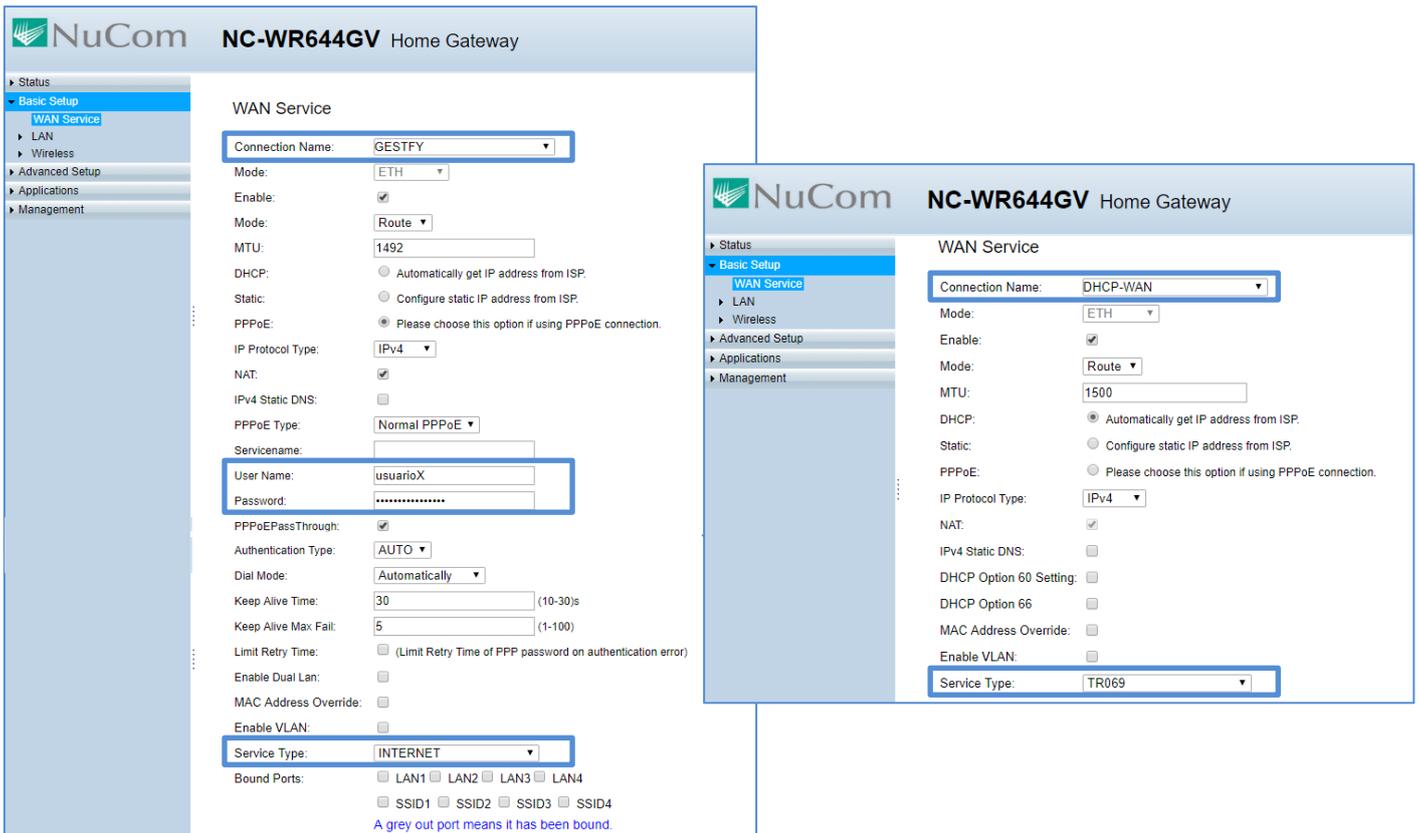


Figura 77 – NuCom Configuración Preset SERIAL – Conexiones WAN

- 5- Por último, el servidor ACS aporta al dispositivo CPE la configuración del último **Preset SPARK**. Este Preset se utiliza para iniciar la conexión **PPPoE** y cambiar el **PeriodicInformInterval** para que el dispositivo realice informes cada 2 minutos. Con este Preset el dispositivo CPE ya tiene conexión a **Internet**.

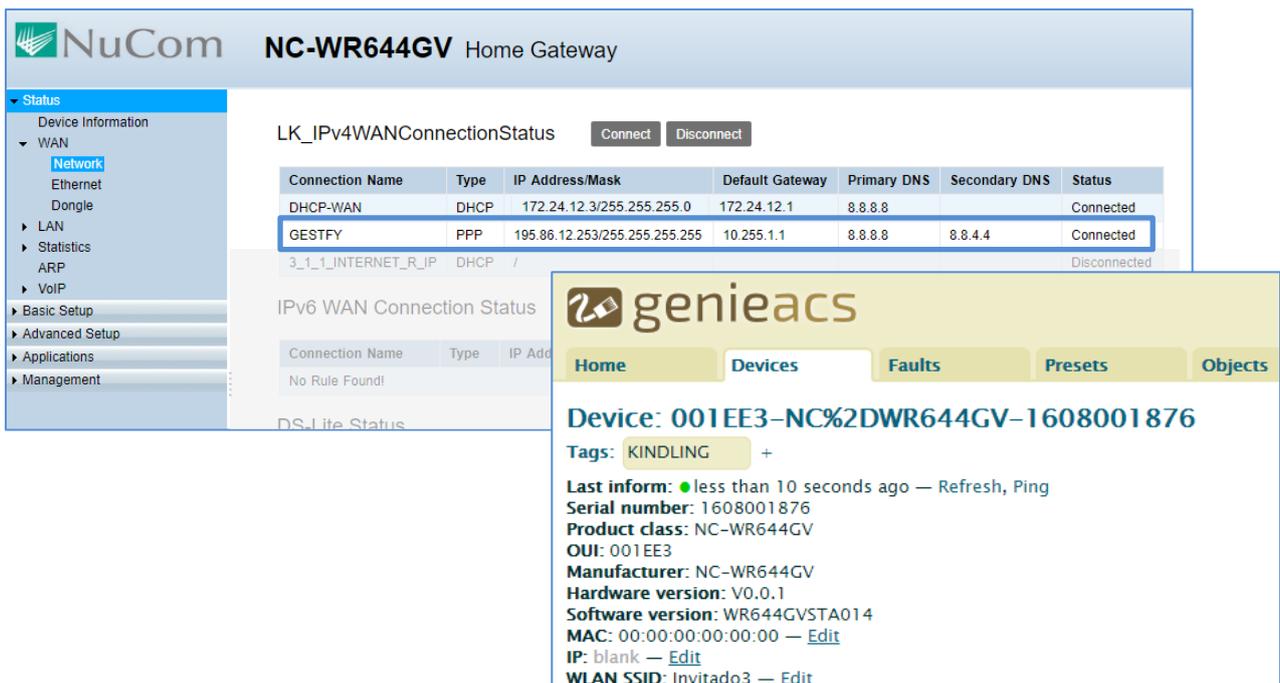


Figura 78 – NuCom Configuración Preset SPARK

3.3.2. Redes FTTH

En las redes FTTH un dispositivo CPE pasa por los siguientes pasos de configuración:

- 1- El **dispositivo CPE** se conecta a la red ISP mediante una **OLT** (Optical Line Terminal). Este dispositivo CPE pide un direccionamiento IP mediante DHCP y recoge información configurada por la OLT. Previamente creamos en la OLT una configuración base para cada fabricante, estas **configuraciones básicas** contienen información sobre las **VLANs** que vamos a aprovisionar al dispositivo CPE además de la dirección IP de nuestro servidor ACS. Realiza este aprovisionamiento mediante el protocolo **OMCI**.

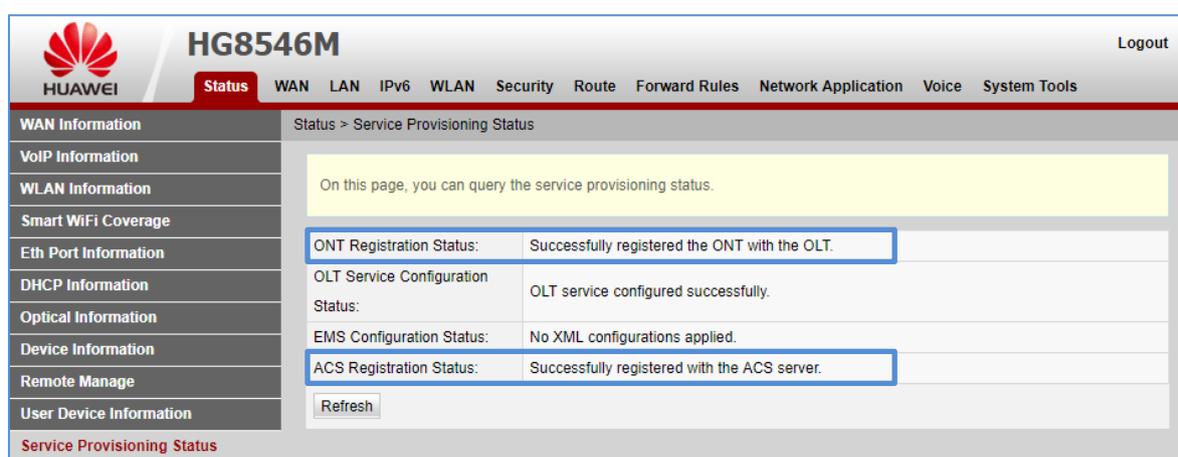


Figura 79 – Huawei aprovisionamiento OLT

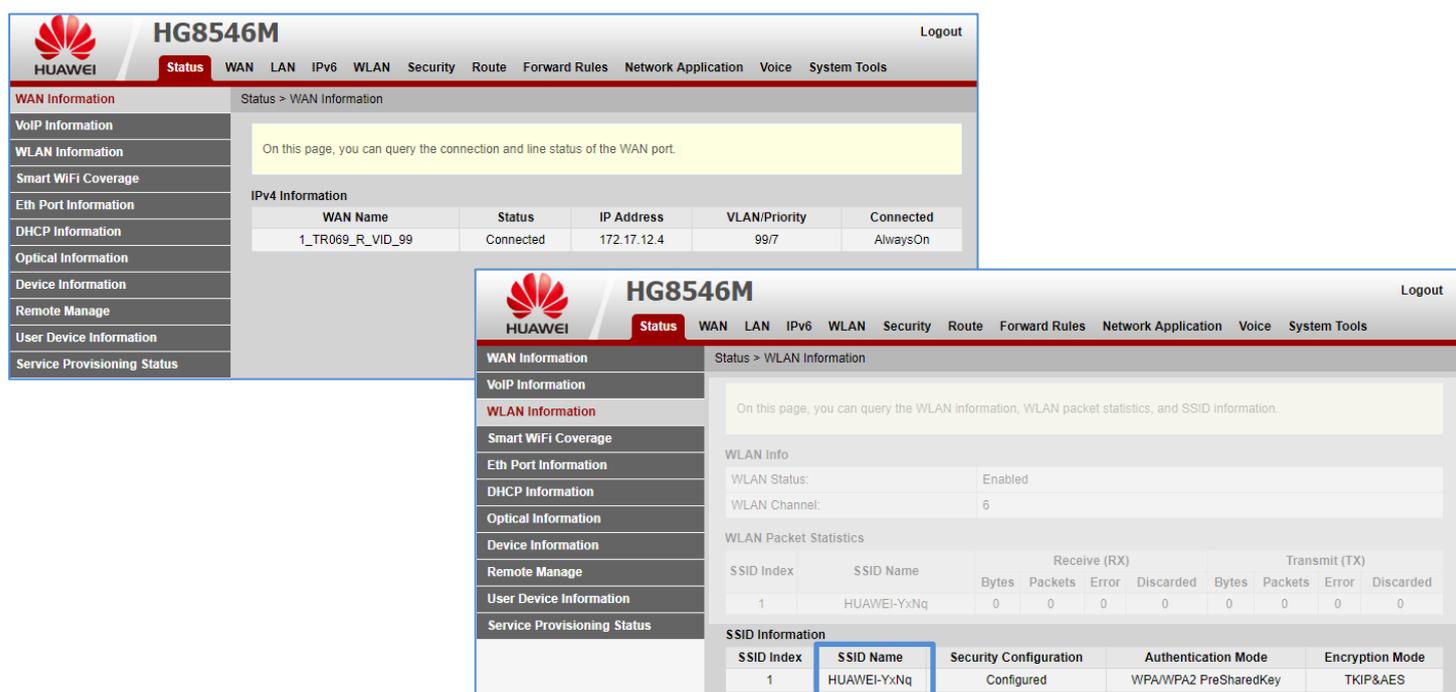


Figura 80 – Huawei configuración WAN y WLAN inicial

- 2- El servidor ACS, una vez realiza conexión con el dispositivo CPE, empieza enviando la **función RPC “RefreshObject”**, para recibir la configuración actual del dispositivo CPE. Cuando ya tenga el dispositivo CPE en su base de datos, el servidor ACS comprueba si el dispositivo cumple las **precondiciones** de los presets configurados. El servidor ACS envía al dispositivo CPE la configuración del **Preset “BIGBANG”**.

The image shows two overlapping screenshots from a Huawei HG8546M web interface. The background screenshot displays the 'System Tools' configuration page for 'ACS Parameter Settings'. It includes fields for 'Enable Periodic Informing' (checked), 'Informing Interval' (10), 'Informing Time', 'ACS URL' (http://192.168.254.2:7547), 'ACS User Name' (admin), and 'ACS Password'. A blue box highlights the 'Connection Request User Name' (usuarioX) and 'Connection Request Password' fields. The foreground screenshot shows the 'genieacs' interface for a device with ID '00259E-HG8546M-485754431B746686'. It lists tags as 'BIGBANG' and provides details such as 'Last inform: half a minute ago', 'Serial number: 485754431B746686', 'Product class: HG8546M', 'OUI: 00259E', 'Manufacturer: Huawei Technologies Co., Ltd', 'Hardware version: 767.A', 'Software version: V3R016C10S130', 'MAC: 4C:F9:5D:1B:74:67', 'IP: 172.17.12.4', and 'WLAN SSID: HUAWEI-YxNq'.

Figura 81 – Huawei configuración Preset BIGBANG

- 3- Una vez el dispositivo CPE tenga la configuración del Preset BIGBANG cumple las precondiciones para el **Preset GENERAL**. En este Preset GENERAL el dispositivo CPE recibe un **objeto PPP** que contiene un conjunto de parámetros genéricos para **conexiones PPP** para el fabricante Huawei.

The image shows a screenshot of the 'genieacs' interface for a device with ID '00259E-HG8546M-485754431B746686'. The 'Tags' section shows 'GENERAL' selected. The interface provides details such as 'Last inform: less than a minute ago', 'Serial number: 485754431B746686', 'Product class: HG8546M', 'OUI: 00259E', 'Manufacturer: Huawei Technologies Co., Ltd', 'Hardware version: 767.A', 'Software version: V3R016C10S130', 'MAC: 4C:F9:5D:1B:74:67', 'IP: 172.17.12.4', and 'WLAN SSID: HUAWEI-YxNq'.

Figura 82 – Huawei configuración Preset GENERAL – Tag GENERAL

The screenshot displays the Huawei HG8546M web management interface. The top navigation bar includes 'Status', 'WAN', 'LAN', 'IPv6', 'WLAN', 'Security', 'Route', 'Forward Rules', 'Network Application', 'Voice', and 'System Tools'. The 'WAN Information' section is active, showing a table of IPv4 connections:

WAN Name	Status	IP Address	VLAN/Priority	Connected
1_TR069_R_VID_99	Connected	172.17.12.4	99/7	AlwaysOn
GESTFY	Disconnected	--	1/0	AlwaysOn

The configuration window for the 'GESTFY' connection shows the following settings:

- Basic Information:**
 - Enable WAN:
 - Encapsulation Mode: IPoE PPPoE
 - Protocol Type: IPv4
 - WAN Mode: Route WAN
 - Service Type: INTERNET
 - Enable VLAN:
 - VLAN ID: 1 (range 1-4094)
 - 802.1p Policy: Use the specified value Copy from IP precedence
 - 802.1p: 0
 - MRU: 1492 (range 1-1540)
 - User Name: [empty]
 - Password: [masked]
 - Enable LCP Detection:
 - Binding Options: LAN1 LAN2 LAN3 LAN4 SSID1 SSID2 SSID3 SSID4
- IPv4 Information:**
 - IP Acquisition Mode: Static DHCP PPPoE

Figura 83 – Huawei configuración Preset GENERAL – Objeto PPP

- 4- El dispositivo CPE, cuando ya tenga el objeto PPP y el TAG “GENERAL”, cumple los prerequisites para el **Preset SERIAL**. El servidor ACS envía la **configuración específica** del cliente, determinado por el número de serial del CPE, para su conexión a Internet con **PPPoE**. También se configura las conexiones WAN de manera que la conexión **DHCP** se utiliza para las comunicaciones **TR-069** y **PPPoE** para la conexión a **Internet**.

The screenshot shows the genieacs web interface. The navigation bar includes 'Home', 'Devices', 'Faults', 'Presets', and 'Objects'. The main content area displays the following information:

- Device:** 00259E-HG8546M-485754431B746686
- Tags:** GENERAL SERIAL +
- Last inform:** less than 10 seconds ago — Refresh, Ping
- Serial number:** 485754431B746686
- Product class:** HG8546M
- OUI:** 00259E
- Manufacturer:** Huawei Technologies Co., Ltd
- Hardware version:** 767.A
- Software version:** V3R016C10S130
- MAC:** 4C:F9:5D:1B:74:67
- IP:** 172.17.12.4 — Edit
- WLAN SSID:** gestfylab — Edit

Figura 84 – Huawei configuración Preset SERIAL – Tag SERIAL

The screenshot shows the Huawei HG8546M configuration interface. The top navigation bar includes: Status, WAN, LAN, IPv6, WLAN, Security, Route, Forward Rules, Network Application, Voice, System Tools. The main content area is titled 'Status > WAN Information' and contains a table of WAN connections:

WAN Name	Status	IP Address	VLAN/Priority	Connected
1_TR069_R_VID_99	Connected	172.17.12.4	99/7	AlwaysOn
GESTFY	Disconnected	--	10/0	AlwaysOn

Below the table, two configuration panels are shown side-by-side:

- Left Panel (Connection: 1_TR069_R_VID_99):**
 - Basic Information: Enable WAN (checked), Encapsulation Mode (IPoE), Protocol Type (IPv4), WAN Mode (Route WAN), Service Type (TR069), Enable VLAN (checked), VLAN ID (99), 802.1p Policy (Use the specified value), 802.1p (7), MTU (1500).
 - IPv4 Information: IP Acquisition Mode (DHCP).
- Right Panel (Connection: 1_TR069_R_VID_99):**
 - Basic Information: Enable WAN (checked), Encapsulation Mode (PPPoE), Protocol Type (IPv4), WAN Mode (Route WAN), Service Type (INTERNET), Enable VLAN (checked), VLAN ID (10), 802.1p Policy (Use the specified value), 802.1p (0), MRU (1492), User Name (uz3ww08a), Password (masked), Enable LCP Detection (unchecked), Binding Options (LAN1, LAN2, LAN3, LAN4, SSID1, SSID2, SSID3, SSID4).
 - IPv4 Information: IP Acquisition Mode (PPPoE).

Figura 85 – Huawei configuración Preset SERIAL – Conexiones WAN

- 5- Por último, el servidor ACS aporta al dispositivo CPE la configuración del **Preset SPARK**. Con este Preset se **inicia la conexión PPPoE** configurada y cambia el valor de **PeriodicInformInterval** para que el dispositivo realice informes cada 2 minutos. Con este Preset el dispositivo CPE ya tiene conexión a **Internet**.

The screenshot shows the Huawei HG8546M configuration interface. The top navigation bar includes: Status, WAN, LAN, IPv6, WLAN, Security, Route, Forward Rules, Network Application, Voice, System Tools. The main content area is titled 'Status > WAN Information' and contains a table of WAN connections:

WAN Name	Status	IP Address	VLAN/Priority	Connected
1_TR069_R_VID_99	Connected	172.17.12.4	99/7	AlwaysOn
GESTFY	Connected	195.86.12.252	10/0	AlwaysOn

Below the table, a **genieacs** device management panel is shown:

- Navigation: Home, Devices, Faults, Presets, Objects
- Device: 00259E-HG8546M-485754431B746686
- Tags: KINDLING +
- Last inform: half a minute ago — Refresh, Ping
- Serial number: 485754431B746686
- Product class: HG8546M
- OUI: 00259E
- Manufacturer: Huawei Technologies Co., Ltd
- Hardware version: 767.A
- Software version: V3R016C10S130
- MAC: 4C:F9:5D:1B:74:67
- IP: 172.17.12.4 — Edit
- WLAN SSID: gestfylab — Edit

Figura 86 – Huawei configuración Preset SPARK

3.4. Gestfy

Gestfy es un **entorno web** intuitivo y sencillo, creado por la empresa ISP donde se realizaron las prácticas, que permite **gestionar** de la forma más eficaz un ISP. Implementa **funcionalidades** como facturación de cuotas, RADIUS, portal de cliente, etc.

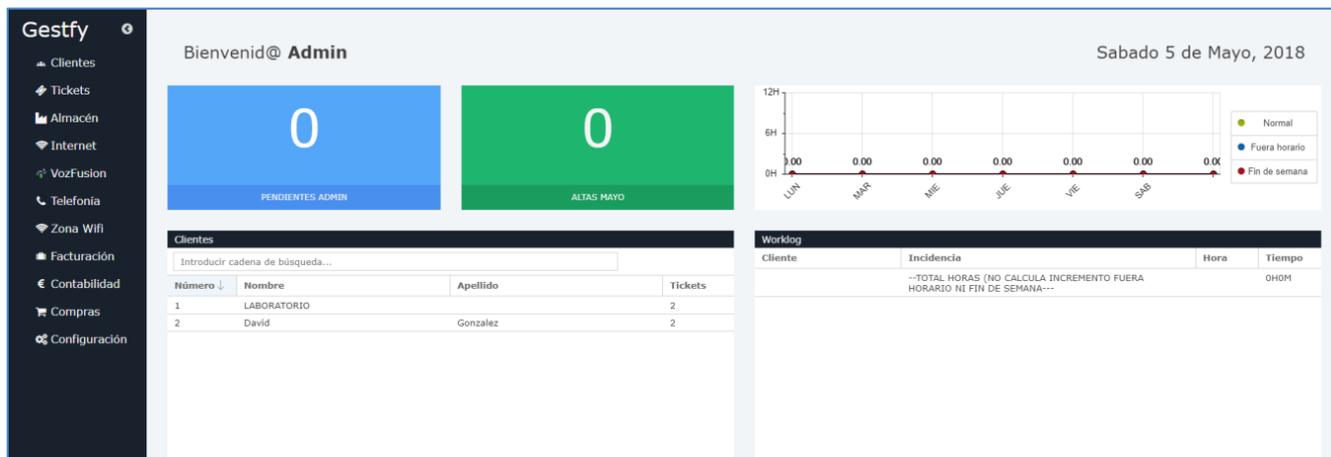


Figura 87 – Gestfy página principal

Además de estas funcionalidades, junto con la empresa, hemos implementado el servicio desarrollado durante las prácticas, debido a que el sistema ACS es un servicio muy práctico para un ISP.

Una vez se configuró **GenieACS** y observamos que todo funciona empezamos a implementar las funciones **TR-069** de GenieACS en Gestfy. El objetivo era facilitar a los técnicos de la empresa configurar los Presets.

Primero se diseñó una pestaña para **crear y modificar los Presets** que pueda tener el sistema ACS. En esta pestaña especificamos un Nombre, una Descripción y los Parámetros que va a tener dicho Preset.

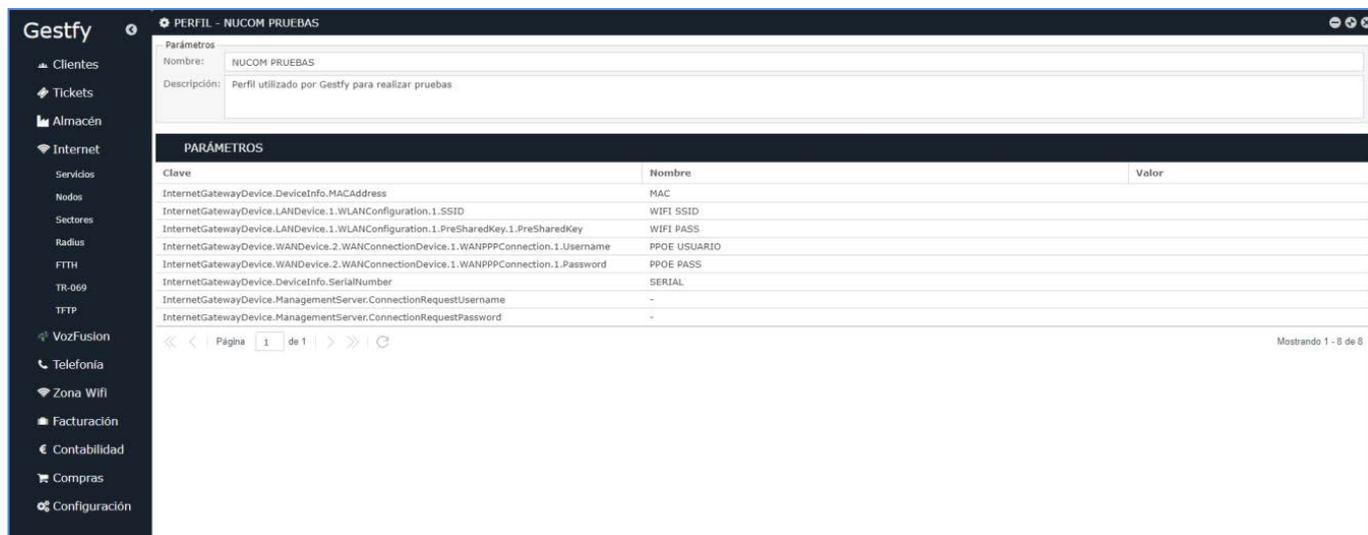


Figura 88 – Gestfy Pestaña Presets

Una vez definido los Presets, se puede el Preset Serial configurar para cada cliente. Dentro de la pestaña de un cliente, donde se encuentran sus datos de cliente, observamos la Pestaña para definir los campos del Preset Serial. De esta manera tenemos todos los datos de un cliente agrupado.

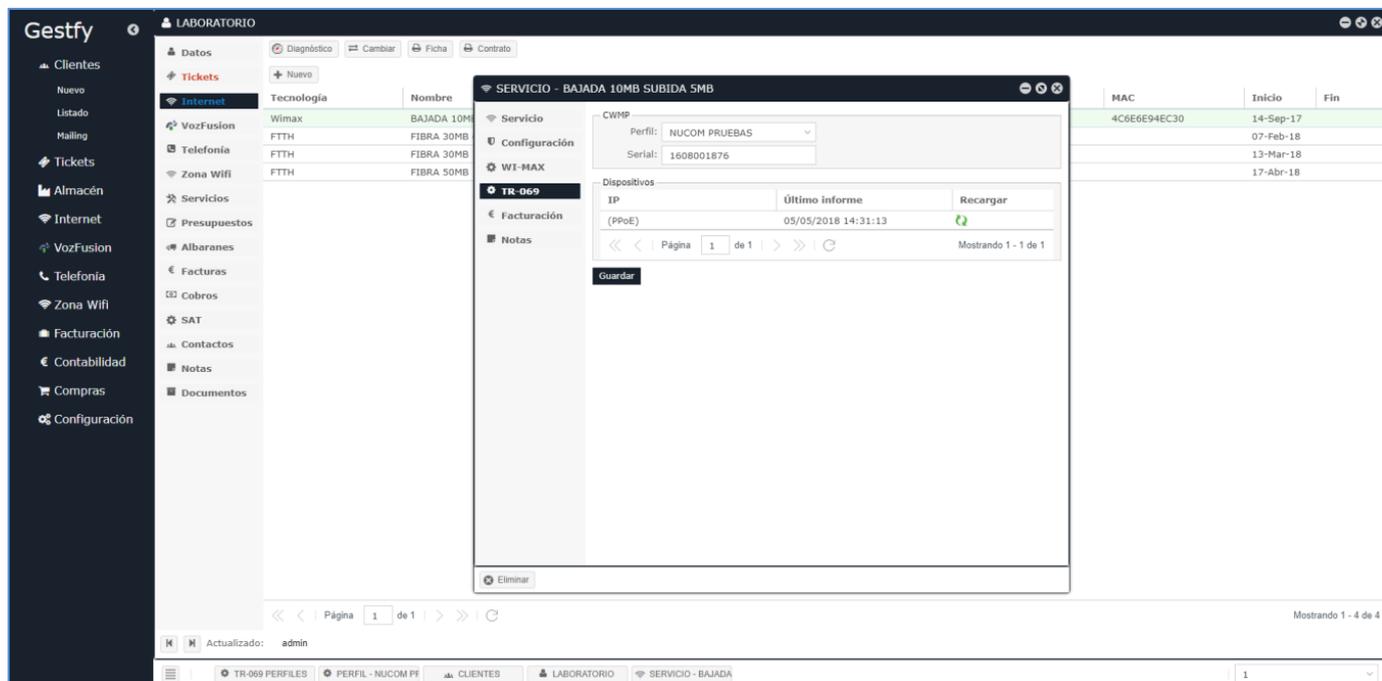


Figura 89 – Gestfy Pestaña Cliente – TR069

4. Conclusiones y Futuras líneas de Trabajo

El **objetivo** de este proyecto fin de grado era conseguir un sistema capaz de **aprovisionar** archivos de configuración a dispositivos CPE de manera **automática** y poder monitorizar dichos dispositivos (Auto-Configuration Server - **ACS**).

*Para alcanzar este objetivo empezamos a investigar sobre las posibles soluciones ACS. Entre las soluciones que encontramos decidimos utilizar **GenieACS**. Una vez decidido el sistema ACS, empezamos a instalar GenieACS sobre un servidor de pruebas de la empresa (**HP Proliant Microserver**).*

*Diseñamos una **red de pruebas** para simular como funcionaría el sistema en un ISP. Esta red se diseñó con las tecnologías **FTTH y Wireless/WIMAX** para observar el comportamiento del sistema ante ambos casos. Una vez montada la red de pruebas con el servidor empezamos a configurar GenieACS, creando **Presets** para aprovisionar a los dispositivos CPE.*

*Por último, para la tecnología **Wireless/WIMAX** instalamos un **servidor TFTP** para aprovisionar, en primera instancia, a un dispositivo CPE la dirección IP del Servidor ACS. Mientras que, por otro lado, la tecnología **FTTH**, al disponer de un protocolo propio para el aprovisionamiento (**OMCI**), le configuramos la dirección IP del servidor ACS sin tener que utilizar el servidor TFTP.*

Con toda esta configuración **logramos nuestro objetivo**, un sistema capaz de aprovisionar dispositivos CPE de un ISP de manera automática independiente de la tecnología empleada en la red del ISP.

Este servicio ACS implica **facilitar** el trabajo a los técnicos a la hora de realizar modificaciones en los **datos de conexión** del cliente, modificando dichos datos desde la plataforma GenieACS. De manera que, un técnico pueda entrar en el **Preset SERIAL** de un cliente, configurarle sus datos de conexión a Internet, llevar el dispositivo CPE al cliente y conectarlo a la red del ISP.

Con esta configuración, el router recibe una **dirección IP por DHCP** y también recibe la dirección IP del **servidor ACS para TR-069**. Una vez tenga la dirección IP del servidor ACS empieza el proceso de auto-aprovisionamiento, cogiendo la configurando de los diferentes Presets dejando finalmente al dispositivo CPE con **acceso a Internet**. Además, con el servicio se puede **monitorizar** los dispositivos CPE de la red ISP, observando el estado del dispositivo si está en línea o desconectada.

Como **futuras mejoras**, durante la fase de configurar las opciones DHCP, para aportar la dirección IP del servidor ACS a los dispositivos CPE, decidimos implementar la **Opción 43 DHCP** antes de la **Opción 66**. Sin embargo, observamos que el dispositivo CPE obtenía su dirección IP pero no la dirección IP del servidor ACS. Resultaba que la dirección IP del servidor ACS que le aprovisionamos si llegaba al dispositivo, pero el dispositivo lo **rechazaba**.

Nos pusimos en contacto con los **fabricantes** del dispositivo que estábamos comprobando y descubrimos que el dispositivo todavía no estaba preparado para recibir datos por la **Opción 43**, pero que es una actualización que tienen previsto para el futuro.

Por tanto, una **futura mejora** sería **implementar Opción 43** en lugar de Opción 66 en cuanto esté configurado bien en los dispositivos CPE para **aumentar seguridad**, eliminando el servidor TFTP.

Otra futura mejora sería **actualizar** la versión de GenieACS a la **versión v1.1**. Con esta versión se implementan funcionalidades como **Scripts para la configuración de Presets**, haciendo que podamos poner condiciones más dinámicas a los Presets. De esta manera podemos **implementar funciones** que detecten cuando un **dispositivo deje de comunicar con GenieACS** y que le vuelva a enviar la configuración de los presets **automáticamente**.

Por último, como **futura mejora**, sería la **implementación** y configuración de **GenieACS en Gestfy**, de manera que podamos diseñar un sistema parecido al **Portal Alejandra de Telefónica**. El Portal Alejandra de Telefónica es un portal de cliente que permite **realizar ciertas modificaciones** a su dispositivo CPE como: modificar la clave de wifi, abrir puertos del router, configurar tu red local, etc. Con **Gestfy**, se podría diseñar un portal de cliente que realiza funciones parecidas mediante el uso de **TR-069**.

5. Bibliografía

Diseño de celdas - (<https://www.comunicacionesinalambricashoy.com/wireless/das-distributed-antennas-systems-sistemas-de-antenas-distribuidas/>); (<https://rfelements.com/>);

ONT - (<http://www.ad-net.com.tw/introduction-to-olt-optical-line-terminal-and-ont-optical-network-terminal-onu-optical-network-unit/>);

AAA - (<https://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-35/101-aaa-part1.html>); (http://apprize.info/network/ccnp_3/2.html);

RADIUS - (<https://www.cisco.com/c/en/us/support/docs/security-vpn/remote-authentication-dial-user-service-radius/12433-32.html>);

Opción 43 - (<https://www.youtube.com/watch?v=Q90rqgLRJd0>);

Opción 66 - (<https://www.incognito.com/tips-and-tutorials/dhcp-options-in-plain-english/>);

CWMP - (<http://qponolution.com/tr-069-basic-fundamental.html>);

SOAP - (https://www.w3schools.com/XML/xml_soap.asp);

TR-098 - (https://www.broadband-forum.org/technical/download/TR-098_Amendment-2.pdf);

Objects - (<https://github.com/genieacs/genieacs/issues/36>);

GenieACS - (<https://genieacs.com/>);

Portal Alejandra - (<https://www.movistar.es/particulares/internet/adsl-fibra-optica/clientes/configuracion-routers-portal-alejandra/>);