UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Metodología para el parcheo de servidores: planificación, optimizaciones y automatización

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Palacios Garcia, Miguel

Tutor/a: Andrés Martínez, David de

Cotutor/a externo: Teruel García, Miguel

CURSO ACADÉMICO: 2023/2024

Agradecimientos

En primer lugar, me gustaría agradecer a toda mi familia, por no haber dudado de mis capacidades en ningún momento y tener fe ciega en mí.

También quiero agradecer a mis amigos, por ayudarme a desconectar en los momentos de agobio.

Me gustaría hacer una mención especial a Irene, por ser un apoyo incondicional día tras día, por ser la persona que mejor me entiende y haber estado tanto en lo bueno como en lo malo.

Por último, me gustaría agradecer enormemente a mi tutor del TFG, David, que desde tercero de carrera fue uno de los mejores profesores que se cruzó en mi camino y ha sido de gran ayuda para la realización de este trabajo. También he de mencionar a Miguel, mi coordinador de prácticas y co-tutor que me integró en el equipo desde el minuto uno.

Resum

Este Treball fi de grau perseguix desenvolupar una metodologia general a seguir per a realitzar correctament els pegats de servidors. Amb esta metodologia es pretén resoldre el problema actual que no existix un document que abaste tots els aspectes fonamentals dels pegats de servidors, a més de no existir un enfocament empresarial en el qual es mostre com s'hauria de tractar amb el client. Amb este treball, es pretén organitzar i gestionar els pegats de servidors d'una forma sistemàtica i de la forma més automatitzada possible, per a un major estalvi de recursos i una major eficiència operativa, aconseguint un enfortiment en l'organització. Aplicant esta metodologia, permetrà mitigar vulnerabilitats i mantindre la seguretat dels sistemes informàtics amb actualitzacions periòdiques, sense afectar la producció o negoci del client, fins i tot en el cas d'enfrontar-nos a un nombre massiu de servidors. Per a aconseguir realitzar esta metodologia correctament, s'han d'explicar i detallar tres aspectes importants: la planificació acordada al costat del client amb tots els factors a considerar, les ferramentes d'automatització possibles i l'execució dels pegats i les seues optimitzacions possibles. Incloent tots estos aspectes s'espera aconseguir una metodologia robusta i completa que servisca de guia i contribuïsca al coneixement en seguretat informàtica.

Paraules clau: pegats, KB, planificació, seguretat, servidor, actualització, vulnerabilitat, Windows Server, reinici, crític, configuració, automatització, instantània, client, còpia de seguretat

Resumen

Este Trabajo Fin de Grado persigue desarrollar una metodología general a seguir para realizar correctamente el parcheo de servidores. Con esta metodología se pretende resolver el problema actual de que no existe un documento que abarque todos los aspectos fundamentales del parcheo de servidores, además de no existir un enfoque empresarial en el que se muestre como se debería tratar con el cliente. Con este trabajo, se pretende organizar y gestionar el parcheo de servidores de una forma sistemática y de la forma más automatizada posible, para un mayor ahorro de recursos y una mayor eficiencia operativa, logrando un fortalecimiento en la organización. Aplicando esta metodología, permitirá mitigar vulnerabilidades y mantener la seguridad de los sistemas informáticos con actualizaciones periódicas, sin afectar la producción o negocio del cliente, incluso en el caso de enfrentarnos a un número masivo de servidores. Para lograr realizar esta metodología correctamente, se deben explicar y detallar tres aspectos importantes: la planificación acordada junto al cliente con todos los factores a considerar, las herramientas de automatización posibles y la ejecución del parcheo y sus optimizaciones posibles. Incluyendo todos estos aspectos se espera lograr una metodología robusta y completa que sirva de guía y contribuya al conocimiento en seguridad informática.

Palabras clave: parcheo, KB, planificación, seguridad, servidor, actualización, vulnerabilidad, Windows Server, reinicio, crítico, configuración, automatización, instantánea, cliente, copia de seguridad

Abstract

This Final Degree Project aims to develop a general methodology to follow in order to correctly patch servers. This methodology aims to solve the current problem that there is no document that covers all the fundamental aspects of server patching, in addition to the lack of a business approach that shows how the client should be treated. With this work, it is intended to organize and manage the patching of servers in a systematic way and in the most automated way possible, for greater savings of resources and greater operational efficiency, achieving a strengthening in the organization. Applying this methodology will allow mitigating vulnerabilities and maintaining the security of computer systems with periodic updates, without affecting the client's production or business, even in the case of facing a massive number of servers. To achieve this methodology correctly, three important aspects must be explained and detailed: the planning agreed with the client with all the factors to be considered, the possible automation tools and the execution of the patch and its possible optimizations. By including all these aspects, it is expected to achieve a robust and complete methodology that serves as a guide and contributes to knowledge in computer security.

Key words: patching, KB, planning, security, server, update, vulnerability, Windows Server, reboot, critical, configuration, automation, snapshot, client, backup

Índice general

Agradecimientos II
Índice general VI
Índice de figuras VIII
Índice de tablas X
Índice de acrónimos XI

1 Introducción 1
1.1 Motivación . . . 2
1.2 Objetivos . . . 3
1.3 Impacto esperado . . . 3
1.4 Metodología . . . 4
1.5 Estructura de la memoria . . . 5
1.6 Convenciones utilizadas . . . 6
2 Estado del arte 7
2.1 Contexto del proyecto . . . 7
2.2 Crítica al estado del arte . . . 7
2.3 Propuesta de mejora . . . 9
3 Planificación y diseño 10
3.1 Planificación con el cliente . . . 11
3.1.1 Credenciales y accesos . . . 11
3.1.2 Equipos implicados . . . 12
3.1.3 Grupos de mantenimiento . . . 13
3.1.4 Ventanas de mantenimiento . . . 13
3.1.5 Uso de herramientas de automatización . . . 14
3.2 Sistemas operativos . . . 14
3.2.1 Versiones de Windows Server . . . 16
3.2.2 Catálogo de Microsoft Update . . . 16
3.2.3 Pilas de servicio y posibles problemas de los parches . . . 18
3.3 Servidores necesarios en una organización . . . 20
3.3.1 Controladores de dominio . . . 20
3.3.2 Servidor de **backup** . . . 20
3.3.3 Servidor de bases de datos . . . 22
3.4 Diversidad tecnológica . . . 23
3.4.1 Hipervisores . . . 23
3.4.2 Infraestructura en la nube . . . 25
3.4.3 Servidores físicos . . . 25
3.5 Plan de emergencia . . . 25
3.5.1 Máquinas virtuales . . . 26
3.5.2 Servidor físico . . . 27
3.5.3 Controladores de Dominio . . . 27
3.5.4 Nodo físico de un *cluster* . . . 28
4 Ejecución y optimizaciones posibles 30

4.1 Previo al parcheo . . . 30
4.1.1 Notificación al cliente . . . 30
4.1.2 Comprobación de espacio de almacenamiento disponible . . . 32
4.1.3 Comprobación previa de servicios . . . 35
4.1.4 Pasos extra . . . 36
4.2 Durante el parcheo . . . 36
4.2.1 Creación de *snapshot* o *backup* . . . 36
4.2.2 Instalación del parche . . . 37
4.2.3 Reinicio del servidor . . . 40
4.3 Posterior al parcheo . . . 42
4.3.1 Comprobación posterior de servicios . . . 42
4.3.2 Comprobación parche instalado . . . 42
4.3.3 Otras comprobaciones . . . 43
4.3.4 Notificación al cliente . . . 43
4.3.5 Resolución de problemas y replanificación del parcheo . . . 44
4.4 Casos especiales . . . 44
4.5 Conclusiones . . . 47

5 Herramientas y automatizaciones posibles **48**
5.1 Herramientas de *ticketing* . . . 48
5.2 Herramientas de administración y gestión de conexiones remotas . . . 51
5.3 Herramientas de monitorización . . . 54
5.4 Herramientas de administración y gestión de parches . . . 56
5.4.1 Con agente . . . 56
5.4.2 Sin agente . . . 60

6 Estudio de caso realista **68**
6.1 Contexto previo . . . 68
6.2 Acuerdo del contrato . . . 68
6.3 Desarrollo . . . 69

7 Conclusiones **73**
7.1 Cumplimiento de objetivos y problemas encontrados . . . 73
7.2 Relación del trabajo desarrollado con los estudios cursados . . . 74
7.3 Trabajos futuros . . . 74

Bibliografía **76**

Apéndice

A Objetivos de Desarrollo Sostenible **79**

Índice de figuras

1.1 Diagrama de Gantt de mi TFG 5

3.1 Diagrama resumen de la planificación y diseño del parcheo 10
3.2 Ticket de rutina de parcheo semestral 11
3.3 Distribución de los 500 superordenadores más potentes del mundo en junio de 2017, por familia de sistemas operativos 14
3.4 Cuota del mercado mundial de servidores por sistema operativo en 2018 y 2019 15
3.5 Cuota de mercado de sistemas operativos de servidor, 2018 en el ámbito empresarial 15
3.6 Catálogo de Microsoft Update para Windows Server 2016 17
3.7 Reemplazo de la actualización 17
3.8 Problemas de filtrado de memoria en el parche de marzo 18
3.9 Documento de seguimiento de los parches 19
3.10 Agente de *backup* de **Veeam** gestionado por el propio servidor 21
3.11 Resumen de la creación de un *backup* para un servidor físico desde **Veeam** 22
3.12 Comparación de uso de hipervisores en 2020 [16] 23
3.13 *Snapshot* de una máquina virtual desde **VMware vCenter** 26

4.1 Diagrama de flujo para la correcta ejecución del parcheo de servidores. La sección en la que se comenta cada uno de los aspectos considerados se identifica en rojo en el diagrama 31
4.2 Correo para notificar al cliente y equipos implicados previo al parcheo de servidores 32
4.3 Limpieza de la unidad **C** 33
4.4 Configuración de perfiles de usuario desde la configuración avanzada del sistema 34
4.5 Eliminar perfiles de usuario 34
4.6 Perfiles temporales desde el editor del registro de Windows 35
4.7 Valor *ProfileImagePath* del usuario que se quiere recuperar el perfil 35
4.8 Comprobación de servicios de Windows desde **Windows Powershell** . . . 36
4.9 *Snapshot* incluyendo la memoria de la máquina virtual 37
4.10 Instalador independiente de Windows Update 38
4.11 Instalación manual de un parche de Windows Update 38
4.12 Opciones mostradas para configurar desde **Sconfig** 39
4.13 Actualización de Windows desde el menú de **Sconfig** 39
4.14 Configuración de la frecuencia de ejecución de una tarea programada . . . 41
4.15 Parámetros y acciones para reiniciar un servidor mediante una tarea programada 41
4.16 Comprobación de servicios mediante el administrador de tareas 42
4.17 Comprobación de la instalación del parche mediante **Windows Powershell** 43
4.18 Correo para notificar al cliente y equipos implicados posterior al parcheo . 44
4.19 *Live Migration* de una máquina virtual desde el *Failover Cluster Manager* . . 45
4.20 Configuración de una máquina virtual para desactivar el inicio automático 45

4.21 Mover una máquina que no está en el *cluster* a otro **Hyper-V** 46
4.22 Mover discos asociados a los nodos del **Hyper-V** 46
4.23 Panel de nodos **Hyper-V** 46
4.24 Pausar el nodo correspondiente y sus roles asociados 47

5.1 Servidores mostrados en la CMDB de la herramienta **SD+** 49
5.2 Cambios preparados para el parcheo de servidores de la herramienta **SD+** 50
5.3 Programación de un cambio de la herramienta **SD+** 50
5.4 Panel de navegación de **RDM** 52
5.5 Listado de credenciales de **RDM**. 52
5.6 Opciones de configuración para el acceso a un servidor en **RDM**. 53
5.7 Permisos de acceso a un servidor en **RDM**. 53
5.8 Características recopiladas de un servidor mediante **Zabbix** 55
5.9 Gráfica del uso de CPU de un servidor mediante **Zabbix** 55
5.10 Apartado de problemas de **Zabbix** 56
5.11 Alerta de reinicio de un servidor en **Zabbix** 56
5.12 Creación de un grupo personalizado desde **Endpoint Central** 57
5.13 Calendario semanal acorde a la división regular de una semana 58
5.14 Calendario semanal acorde al *Patch Tuesday* 58
5.15 Creación de una directiva de implementación desde **Endpoint Central** . . 58
5.16 Actividades posteriores de una directiva de implementación desde **Endpoint Central** 59
5.17 Implementación automática desde **Endpoint Central**. 60
5.18 Resultado de una implementación automática desde **Endpoint Central** . . 60
5.19 Sincronización del servidor **WSUS** con Microsoft Update 61
5.20 GPO configurada para la descarga e instalación de los parches al grupo *Servidores* 62
5.21 Interfaz de **Batchpatch** 63
5.22 Añadir servidores en **Batchpatch** 64
5.23 Obtener más información de un servidor en **Batchpatch** 64
5.24 Crear una nueva implementación desde el apartado de acciones en **Batchpatch** 65
5.25 Crear una implementación para un Windows Server 2012R2 66
5.26 Crear una tarea programada en **Batchpatch** 66
5.27 Ejecutar **Batchpatch** como servicio 67

6.1 Ejemplo de dos cuentas almacenadas en **Syspass** 69
6.2 Configurar la VPN con **FortiClient** 70
6.3 Planificación para el parcheo de servidores de **FarmaLicex** 71
6.4 Creación grupos para el parcheo de servidores de **FarmaLicex** desde **Endpoint Central** 72
6.5 Implementación automática para el parcheo de servidores de **FarmaLicex** desde **Endpoint Central** 72

Índice de tablas

1.1 Top 10 vulnerabilidades 2021 según **OWASP** 2

3.1 Comparación hipervisores Tipo 1 24

5.1 Comparación herramientas de *ticketing* 49
5.2 Comparación herramientas de conexiones remotas 51
5.3 Comparación herramientas de monitorización 54

6.1 Planificación del parcheo de servidores para **FarmaLicex** 71

A.1 Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS) 79

Índice de acrónimos

AD Active Directory

API Application Programming Interface

BITS Background Intelligent Transfer Service

CAB Change Advisory Board

CMDB Configuration Management Data Base

CT Competencia Transversal

DC Domain Controller

DNS Domain Name System

DSRM Directory Services Restore Mode

FPGA Field-Programmable Gate Arrays

FSMO Flexible Single Master Operations

GPO Group Policy Object

HTTP Hypertext Transfer Protocol

HTTPS Hypertext Transfer Protocol Secure

ICMP Internet Control Message Protocol

ILO Integrated Ligths-Out

IP Internet Protocol

KB Knowledge Base

LDAP Lightweight Directory Access Protocol

MFA Multiple-Factor Authentication

MV Máquina Virtual

ODS Objetivos de Desarrollo Sostenible

OWASP Open Web Application Security Project

RDM Remote Desktop Manager

RTO Recovery Time Objective

SNMP Simple Network Management Protocol

SO Sistema Operativo

TFG Trabajo Fin de Grado

TFM Trabajo Fin de Máster

TI Tecnologías de la Información

UEM Unified Endpoint Management

UPV Universidad Politécnica de Valencia

USN Update Sequence Number

VPN Virtual Private Network

WSUS Windows Server Update Services

CAPÍTULO 1
Introducción

En la actualidad, la tecnología ha adoptado un papel omnipresente en nuestro día a día. La informática crece a un ritmo vertiginoso, contemplando cada vez más potencia de computación, mejoras en la inteligencia artificial, y siendo crucial en actividades empresariales y en cualquier ámbito que sea imaginable, haciendo que se tenga una visión optimista para mejorar nuestras vidas y conseguir un futuro más próspero. Sin embargo, esto trae consigo un tema importante a tratar: la seguridad en los sistemas informáticos [1]. Es un pilar fundamental para preservar la integridad, disponibilidad y confidencialidad de los sistemas y de la información [2].

En este ámbito de intentar preservar la seguridad informática aparece el concepto de vulnerabilidades. Una vulnerabilidad en el contexto informático representa un grave problema para la seguridad del sistema. Esta debilidad puede ser debida a errores de configuración, procedimientos deficientes o fallos en el diseño. Los delincuentes cibernéticos suelen aprovechar estas vulnerabilidades, como las presentes en los sistemas operativos, para acceder sin autorización, robar datos sensibles o provocar interrupciones en el sistema, produciendo un impacto crítico en los sistemas operativos, aplicaciones o servicios informáticos [3]. El proyecto abierto de seguridad en aplicaciones web, también conocido como **OWASP**[1], un organismo sin ánimo de lucro dedicado a combatir las vulnerabilidades informáticas, documentando sobre los tipos de vulnerabilidades más críticas en los últimos años. **OWASP** publica la lista con las diez vulnerabilidades más importantes cada cuatro años. La última lista de vulnerabilidades publicada por **OWASP** en 2021 se muestra en la tabla 1.1. En esta clasificación se pueden observar amenazas informáticas más simples como la configuración de seguridad incorrecta, refiriéndose a la configuración de servidores, aplicaciones o sistemas informáticos, dejando brechas de seguridad debido a la falta de atención en detalles, configuraciones inseguras o el acceso a funciones y datos importantes, hasta algunas más complejas como la amenaza de inyección, comprometiendo la integridad y confidencialidad de datos, además de la disponibilidad del sistema, necesitando de un control de accesos adecuado, además de prácticas de codificación segura y validación de entradas [4].

Por todo esto, es fundamental mantener actualizados los sistemas operativos, las aplicaciones y las herramientas de seguridad, ya que estas actualizaciones suelen contener correcciones para las vulnerabilidades descubiertas, además de ser conveniente una inversión en un equipo y herramientas de ciberseguridad para mitigar estos riesgos de una manera efectiva.

El parcheo de servidores fundamentalmente se centra en aplicar actualizaciones de seguridad de sistemas operativos (SO), plataformas o aplicaciones de una forma regular, consiguiendo así mitigar pequeños errores o *bugs* y abordando vulnerabilidades recién

[1] *Open Web Application Security Project*. Consultar en: https://owasp.org/.

descubiertas, consiguiendo tener los servidores actualizados y más resistentes frente a ataques de terceros. Es importante debido al constante cambio y evolución de amenazas informáticas, que sin estas medidas correctivas, daría lugar a dejar los sistemas operativos expuestos a *exploits*[2] y a corromper la estabilidad de estos.

Tabla 1.1: Top 10 vulnerabilidades 2021 según **OWASP**

OWASP TOP 10 (2021)	
Posición	**Vulnerabilidad**
A01	Pérdida de control de acceso
A02	Errores criptográficos
A03	Inyección
A04	Diseño inseguro
A05	Configuración de seguridad incorrecta
A06	Componentes vulnerables y desactualizados
A07	Errores de identificación y autenticación
A08	Errores en el software y en la integridad de los datos
A09	Errores en el registro y monitoreo
A10	Falsificación de solicitudes del lado del servidor (SSRF)

Este trabajo de fin de grado mostrará una metodología acerca de cómo realizar correctamente el parcheo de servidores, empezando por desarrollar la planificación y gestión con el cliente de sus servidores, pasando por el desarrollo de la estrategia del parcheo que usaremos y acabando en aplicar el parche y comprobar su correcto funcionamiento o aplicar acciones correctivas en caso de que se necesitara.

Se propondrá una metodología general en este proyecto para globalizar al máximo en cada uno de los apartados, aunque es inevitable tener que especializar para determinados tipos de servidores, herramientas o contextos.

1.1 Motivación

Durante estos años de carrera, he descubierto diversas tecnologías y herramientas de las que he disfrutado aprendiendo y trabajando, llegando a ser una idea para posibles Trabajos Fin de Grado (TFG), desde mi especialización en ingeniería de computadores descubriendo las *Field-Programmable Gate Arrays* (FPGA) hasta la administración y automatización de redes, desarrollado principalmente durante mi estancia Erasmus en República Checa, pudiendo probar y montar mi propia red.

Tras comenzar mis prácticas de empresa en **Sothis**[3], perteneciente a **Nunsys Group**[4], fui al equipo de Operaciones, principales encargados del parcheo de servidores de sus clientes. Desde el primer momento quedé fascinado con la forma de trabajo y la forma de organizar cientos de servidores, cada uno de ellos realizándose en horarios distintos, con diferentes criticidades, requisitos, versiones del SO...

Empezar con este proyecto me ha permitido salir de la burbuja de la universidad y adentrarme en el mundo laboral, observando toda la interacción, ya sea interdepartamental o con clientes externos, viendo como procesan la información crítica y sensible de muchos clientes en sus servidores dedicados.

[2] Deriva del verbo *to exploit* en inglés y significa aprovechar un error o vulnerabilidad para causar un comportamiento no deseado.

[3] Consultora digital donde realicé mis prácticas. Visitar web en: https://www.sothis.tech/.

[4] https://www.nunsys.com/.

Este proyecto me ha hecho aumentar mi conocimiento acerca de la complejidad inherente a la gestión de los servidores. He aprendido a tratar con *clusters* de servidores para su parcheo y la forma de realización en función de la configuración montada en cada cliente. También me ha permitido entender la criticidad e importancia de realizar correctamente cada paso del parcheo de servidores, ya que no solo estamos tratando con una persona o un ordenador personal, sino con la infraestructura de grandes empresas que tienen su negocio independiente, así como horas de producción que hay que respetar.

Además, la motivación de este proyecto va más allá de entender el mundo laboral. Comprendí la necesidad de crear una metodología general que pusiera fin a discrepancias a la hora de realizar el parcheo y se pudiera seguir una estructura general para realizarlo correctamente de la forma más eficiente y correcta posible. En resumen, la idea con este trabajo es contribuir a la seguridad de entornos informáticos utilizando una técnica proactiva para adaptarse al problema y evitarlo antes de que surja en lugar de hacerlo de una forma reactiva.

1.2 Objetivos

Este TFG, centrado en mis prácticas en empresa, tiene como objetivos principales los siguientes:

1. **Diseñar una metodología integral:** crear una metodología general que englobe todas las fases del proceso de una forma cohesionada. Comenzando por la planificación estratégica, la forma del parcheo, posibles optimizaciones y herramientas de automatización, hasta las comprobaciones posteriores y un caso de prueba.

2. **Desarrollar estrategias de planificación:** establecer unas pautas para saber tratar con el cliente, ver qué parches se van a instalar y si traen problemas asociados que puedan impactar en el funcionamiento o producción.

3. **Mostrar las diferentes formas de agrupación de servidores y casos especiales:** enseñar las distintas formas de agrupación de servidores con su correspondiente actuación, además de casos especiales de servidores.

4. **Explorar herramientas de automatización:** investigar y evaluar el uso de distintas herramientas relevantes para el parcheo de servidores, mostrando sus puntos débiles y puntos fuertes. Mostrando la ejecución eficiente del parcheo, además de su control previo y posterior.

5. **Realizar un caso real con todas las variables:** realizar un caso real desde cero, mostrando todas las fases del proceso.

6. **Contribuir al conocimiento en seguridad informática:** que este proyecto sirva como un manual para la gente que se introduce al parcheo de servidores, además de aportar conocimientos prácticos y novedosos en el área de la seguridad informática, compartiendo hallazgos interesantes o lecciones aprendidas, que sean de uso para otras organizaciones o profesionales del sector.

1.3 Impacto esperado

Disponer de una metodología general para el parcheo de servidores puede ser de gran beneficio y reportar un impacto en las siguientes áreas:

1. **Mejora de la seguridad informática:** si se realiza correctamente el trabajo, se conseguirá de manera directa e inmediata un mayor nivel de seguridad en los servidores, además de reducir posibles vulnerabilidades.

2. **Fortalecimiento en la organización:** con esta metodología se logrará mejorar la seguridad en los servidores de las empresas. Logrando empresas más robustas y fuertes ante posibles amenazas.

3. **Mayor eficiencia operativa:** servirá para una mejor estructuración de la información, reducirá tiempos de inactividad y mejorará la comunicación en la organización en lo referente al tema de las actualizaciones de seguridad.

4. **Ahorro de recursos:** con las posibles automatizaciones del proyecto y la optimización de los procesos, se conseguirá un significante ahorro de recursos, principalmente en tiempo empleado por la mano de obra.

5. **Aplicabilidad a diversos sectores:** al ser una metodología que ofrecerá diferentes herramientas y recursos, se podrá adaptar desde pequeñas organizaciones con pocos servidores hasta grandes corporativas con un alto volumen de servidores.

   El impacto esperado de la metodología va acorde a solventar necesidades contemporáneas de un proceso crítico en entornos de TI[5], resaltando en la importancia de la planificación de las actualizaciones. Además, este trabajo es vinculable a varios puntos de los Objetivos de Desarrollo Sostenible[6] (ODS) de las **Naciones Unidas**, estando altamente conectado con el Objetivo 8: *Trabajo decente y crecimiento económico,* al conseguir un ahorro de recursos y tiempo dentro de la organización promoviendo así el crecimiento económico.

1.4 Metodología

Para lograr cumplir todos los objetivos de este proyecto y poder realizar una metodología completa para el parcheo de servidores, se ha decidido usar un diagrama de Gantt para poder fijar los tiempos correctamente y ceñirse al plan establecido. Para realizar correctamente el TFG, estas son las secciones que se han fijado en el diagrama de Gantt de la figura 1.1:

1. **Desarrollo estructuración TFG:** en este apartado se van a dedicar dos semanas a la investigación y al planteamiento de como realizar este proyecto correctamente.

2. **Introducción y objetivos básicos:** se va a pensar, escribir y corregir la introducción y los objetivos en tres semanas, teniendo una semana para cada tarea.

3. **Estado del arte:** se va a realizar de la misma forma que el punto anterior.

4. **Planificación y diseño:** se le van a dedicar seis semanas a la planificación y diseño del TFG, necesitando tres semanas para investigar todos los aspectos posibles y necesarios para poder realizar la planificación de la metodología correctamente.

5. **Ejecución y optimizaciones posibles:** en tres semanas se va a redactar el apartado de como realizar la ejecución sobre el parcheo de servidores.

[5] Tecnologías de la Información.
[6] https://www.undp.org/es/sustainable-development-goals.

6. **Herramientas de automatización**: otro capítulo que va a abarcar una gran extensión de tiempo. Esto es debido a la necesidad de documentarse acerca de todas las herramientas posibles para los distintos apartados del trabajo.

7. **Estudio de caso real:** en dos semanas se va a realizar el estudio de caso real.

8. **Conclusiones:** en dos semanas se van a redactar y corregir las conclusiones del TFG.

9. **ODS y revisiones extra:** por último y a la par que las conclusiones, se van a redactar los ODS y corregir los flecos pendientes que queden del trabajo.

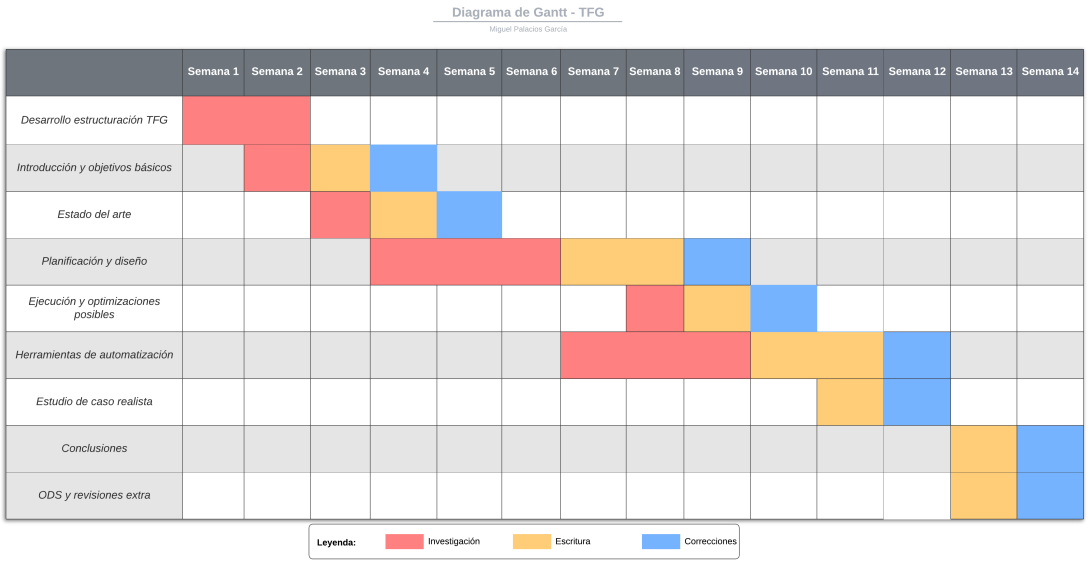


Figura 1.1: Diagrama de Gantt de mi TFG

En el diagrama de Gantt de la figura 1.1 se puede observar cómo varios de los pasos descritos anteriormente se superponen y mientras se investiga sobre algún tema o apartado se va escribiendo o corrigiendo otro, permitiendo mayor flexibilidad y agilidad para trabajar. Gracias a la realización de este diagrama, se logra mantener una estructura para el desarrollo del trabajo.

1.5 Estructura de la memoria

La memoria de este TFG está estructurada en varios capítulos, los cuales se van a detallar a continuación:

1. **Introducción:** se realiza una presentación del tema y se exponen principalmente la motivación y los objetivos a cumplir referentes a este proyecto.

2. **Estado del arte:** se expone el contexto tecnológico actual, los problemas actuales y se presenta brevemente la propuesta de mejora.

3. **Planificación y diseño:** se detalla cómo se planifica y diseña el parcheo de servidores con todas las posibles variantes que esto conlleva.

4. **Ejecución y optimizaciones posibles:** en esta sección, se narran los pasos en la ejecución de la solución, con los posibles inconvenientes que pueden surgir durante el proceso y las formas de optimización de estos.
5. **Herramientas y automatizaciones posibles:** se muestran las herramientas y formas de automatización a la hora de realizar el parcheo, su configuración y todas sus capacidades.
6. **Estudio de caso realista:** en este apartado, se ejemplica un caso realista con todos los pasos seguidos y ordenados secuencialmente para ver el proceso al completo.
7. **Conclusiones:** por último, se hace una recapitulación del trabajo, se comprobarán si los objetivos han sido alcanzados, se revisará lo aprendido durante el proceso y su paralelismo con lo estudiado durante el grado de Ingeniería Informática de la Universidad Politécnica de Valencia (UPV).

1.6 Convenciones utilizadas

Para que sea más amena y sencilla la lectura de este trabajo, se van a definir una serie de convenciones que se van a utilizar a lo largo de todo el documento:

- Las palabras o expresiones extranjeras se escribirán en cursiva.
- Las herramientas, fabricantes y empresas mencionadas se escribirán en negrita.
- Se dispone de un índice de acrónimos para su consulta.

CAPÍTULO 2
Estado del arte

En este capítulo, se va a explorar la situación actual de la tecnología referente al parcheo de servidores, previo a la metodología general que se propone con esta solución.

2.1 Contexto del proyecto

Este TFG se ha realizado en el ámbito de prácticas de empresa, por esto es necesario situar la empresa en el marco del trabajo para que todo se entienda mejor. La empresa en cuestión es **Sothis**, una consultora informática centrada en la transformación digital de sus clientes, pertenece actualmente al grupo **Nunsys Group**. **Sothis** fue fundada en 2008 y su principal fuente de inversión fue **Juan Roig**, a través de su sociedad de inversión **Angels Capital** [5]. En 2022, la empresa fue comprada por **Nunsys**, siendo **Francisco Gavilán** su dueño actual. La empresa tecnológica es líder en diversas áreas como ciberseguridad o SAP. Actualmente, cuenta con más de 400 clientes en 35 países [6].

En relación con este proyecto, la empresa se encarga de la monitorización, mantenimiento y actualización de los servidores y equipos de los clientes que contraten este servicio, lo que lleva a la aplicación de parches de seguridad. En este apartado con la solución propuesta se pretende mejorar y agilizar las tareas de parcheo de servidores, permitiendo a los equipos de dentro de la organización disponer de tiempo para otras tareas de TI.

2.2 Crítica al estado del arte

Actualmente, se dispone de bastante información respecto al parcheo de servidores. A continuación, voy a profundizar en puntos donde la información de la que se dispone es bastante correcta y coherente.

Existen diversos aspectos fundamentales para que el parcheo de servidores se realice de forma correcta, y todos estos aspectos deben seguir un orden en concreto.

En primer lugar, es fundamental saber los equipos implicados en el parcheo de servidores para que el flujo de información sea lo más certero y rápido posible. Los diversos equipos implicados deben estar todos sincronizados: el equipo de ciberseguridad debe detectar vulnerabilidades y transmitirlas al equipo de sistemas para que valoren si tiene una verdadera afectación a los servidores de la organización y transmitirle al equipo encargado del parcheo de servidores, en este caso llamado equipo de operaciones, qué activos y qué parches se deben aplicar, hasta avisar al equipo de monitorización para que sean conscientes de posibles alertas o problemas durante la ejecución del parcheo. Ade-

más, si se están parcheando servidores de terceros, también se requerirá de la comunicación con el cliente. En la literatura revisada, sin embargo, no se especifican claramente los grupos o equipos implicados tanto de forma activa o pasiva o si se requiere de terceros [7][8][10]. En el único artículo que se especifica bien esta relación es en el de *Agile Security Patching* donde se establece la jerarquía de los interesados y miembros participantes a lo largo de todo el proceso [10].

Cuando los equipos implicados ya están creados y coordinados, lo siguiente es establecer los activos que se van a parchear, para este paso es muy importante contar con un inventario actualizado de todos los activos dentro de la organización. Esos activos, en todos los artículos o libros que he revisado, mencionan la importancia de separarlos según su criticidad dentro de la organización y la disponibilidad de cada uno de ellos. En otras palabras, no es lo mismo parchear un servidor de choque[7], que un servidor con ciertos servicios activos en la organización o un activo que esté en producción y no se pueda detener porque causaría grandes pérdidas. El siguiente factor a tratar es establecer que parches van a recibir los activos seleccionados anteriormente. En la literatura revisada, se hacen separaciones según su criticidad, empezando desde parches rutinarios que se aplican de una forma rutinaria cada cierto tiempo, hasta parches de emergencia que hayan salido por alguna vulnerabilidad o mitigaciones temporales si no hay ningún parche que pueda afrontar cierta vulnerabilidad [7][8][9]. Todo lo mencionado anteriormente, aunque ya exista su definición y clasificación, se detallará y definirá debido a su importancia para realizar una metodología robusta y coherente.

Antes de la aplicación de los parches, se requiere tener un plan de emergencia o plan de marcha atrás en caso de que hubiera problemas durante o posterior al parcheo. En los artículos se comenta que se ha de especificar un guión para actuar posteriormente, pero en ninguno se detallan unos posibles pasos a seguir, especialmente para servidores críticos o que requieren de unos pasos adicionales para su recuperación.

Para la aplicación de los parches en los servidores, existe la opción de aplicarlos de forma manual o de usar ciertas herramientas de automatización para agilizar la tarea. En la literatura se mencionan varias herramientas y se dan muestras de su uso, habiendo algunas que se pueden llegar a ver algo desfasadas en la era tecnológica actual u otras siendo demasiadas complejas [9]. Asimismo, para optimizar los procesos se propone el uso de metodologías ágiles para llevar a cabo las actualizaciones y revisiones necesarias [10].

Estos pasos que se han destacado hasta ahora son aplicables de una forma general para cualquier tecnología y el parcheo de cualquier dispositivo, pero es necesario tener en cuenta información concreta para el caso particular del parcheo de servidores haciendo hincapié en herramientas o páginas donde poder ver vulnerabilidades existentes y estar al tanto de las más novedosas. Por ejemplo, cuando se mencionan el tema de vulnerabilidades, la mayoría de escritos abarcan desde dispositivos móviles a equipos de trabajo o equipos de comunicaciones como *routers* o *switches* [8]. También se echa en falta que se hable más de servidores específicos, que requieren un plan de acción diferente, como puede ser un controlador de dominio o un nodo de un *cluster*, ya que estos puntos no se especifican en ningún artículo tratado.

El gran apartado que queda sin resolver y prácticamente ni se menciona, es enfocar el parcheo de servidores desde la visión de un equipo externo, como puede ser una consultora para ofrecer esos servicios a una empresa.

Por último, se han revisado TFG y Trabajos Fin de Máster (TFM) sobre metodologías existentes acerca del parcheo de servidores o la seguridad en los sistemas informáticos.

[7] Servidor diseñador para hacer pruebas o tests.

En algunos TFG se mencionan herramientas de virtualización o configuraciones que se van a mostrar en este trabajo, aunque con un enfoque totalmente distinto. Lo más parecido es un TFM sobre una metodología para la seguridad de los sistemas informáticos donde sí que se establecen una serie de directivas que se deben seguir, como la necesidad de tener un inventario de los activos dentro de la organización, la necesidad de realizar copias de seguridad periódicas, o la mitigación de vulnerabilidades [11]. Estos puntos también serán tratados en este trabajo.

2.3 Propuesta de mejora

Tras el análisis sobre el estado del arte actual, en este punto se van a redactar varias propuestas de mejora basándose en los problemas y aspectos sin resolver del punto anterior:

En primer lugar, se especificarán claramente las divisiones tanto de activos en función de su importancia dentro de la organización como a nivel de producción. También se detallará la diferencia entre las distintas criticidades de los parches y la correcta forma de actuación en cada caso.

En segundo lugar, se hará una clasificación de posibles herramientas de automatización con agente y sin agente, además de mostrar minuciosamente la configuración de las aplicaciones y su uso.

En tercer lugar, se establecerán claramente los roles implicados a lo largo de todo el proceso, añadiendo también variantes en los equipos implicados en función de las necesidades.

En cuarto lugar, en este trabajo se detallará el plan de emergencia o plan de marcha atrás de una forma concisa, diferenciando los distintos grupos de activos en los que hay que trazar planes distintos.

En quinto lugar, se especificará un plan de acción para servidores característicos como el nodo de un *cluster*.

En sexto lugar, tras la nula información sobre enfocar el parcheo de servidores desde la visión de un equipo externo, se detallarán pasos extra que se tendrán que tener en cuenta en estos casos.

Por último, es notoria la falta de información acerca del parcheo únicamente de servidores, por lo que con este trabajo se espera, además de destacar ideas existentes a día de hoy, crear un documento que sea coherente, compacto y unificado con todos los puntos que se deben tratar.

CAPÍTULO 3
Planificación y diseño

En este capítulo se introducirá el parcheo de servidores, detallando y explicando todos los factores, tanto humanos como informáticos, que se deben tener en cuenta a la hora de realizar este proceso.

La primera etapa de todas es la planificación con el cliente, donde surge una estrecha colaboración para discutir todos los requisitos, expectativas y objetivos para el parcheo de los servidores. Se tratarán puntos como las credenciales necesarias para acceder a sus equipos, se establecerán grupos de servidores en función de las necesidades del cliente, además de fijar ventanas de indisponibilidad para el parcheo.

Al separar los activos en los grupos de mantenimiento se debe realizar un inventario de sus características, incluyendo su sistema operativo, aunque para este trabajo solo se va a explicar el parcheo para el sistema operativo **Windows Server**, donde se van a explicar sus características y sus actualizaciones lanzadas por **Microsoft**, el cliente también puede disponer de otros sistemas operativos como **Linux** o **Unix**.

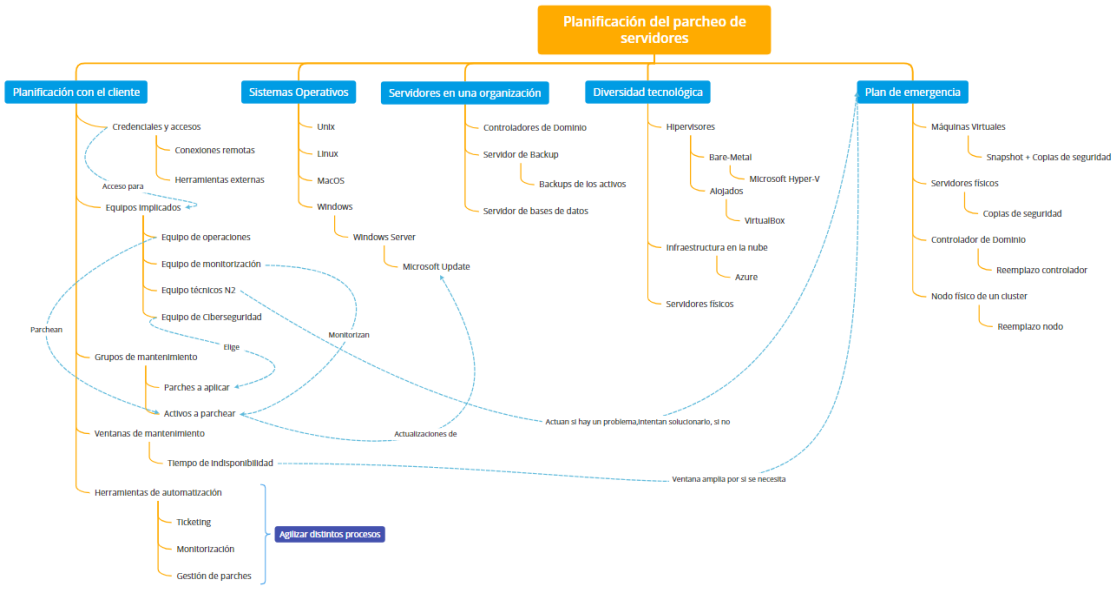


Figura 3.1: Diagrama resumen de la planificación y diseño del parcheo

Ya no solo se debe prestar atención al sistema operativo, sino a la diversidad tecnológica que puede haber dentro de la organización, puesto que el parcheo de un nodo de hipervisor, una máquina virtual alojada en **Azure** o un controlador de dominio van a llevar no solo procesos diferentes para realizar el parcheo correctamente, sino que también

requerirán de un plan de marcha atrás o un plan de emergencia específico por si hubiera problemas durante el parcheo.

En la figura 3.1 se puede observar un diagrama a modo de resumen de los puntos que se van a explicar a continuación.

3.1 Planificación con el cliente

En primer lugar, es importante la planificación con el cliente. Al fin y al cabo, es un servicio destinado para la seguridad de sus servidores y equipos, por lo que lo más aconsejable es tener contacto directo y constante durante todas las fases de la implementación.

Es importante destacar el factor de la periodicidad con la que realizar estas actualizaciones, dependiendo de las necesidades del cliente y de la criticidad de sus servidores, pudiendo ser desde una rutina mensual, trimestral hasta semestral en caso de que sea más por mantenimientos preventivos.

La periodicidad en la que se parchean los servidores se estipula en el contrato que se ha firmado con el cliente. Una vez está establecida esa periodicidad, el equipo resolutor de los parcheos, acordará con el cliente el horario con menor impacto para su infraestructura. De esta forma, se podrá crear una rutina, con una plataforma informática que facilite la gestión de incidencias y peticiones de servicio tanto internas como externas de una manera eficiente y rápida, estas herramientas son conocidas como sistema de *ticketing* como la mostrada en la figura 3.2 que es **ManageEngine ServiceDesk Plus**[8], para que cuando se necesite realizar de nuevo ese parcheo se nos avise. Una herramienta de gestión de servicios de TI es muy recomendable debido a que permite agilizar procesos a la hora de la creación de tareas, rutinas o incidencias de una forma más automatizada. Más adelante se comentarán varias herramientas útiles de este tipo.

En caso de no poder contar con una aplicación como la mencionada, la creación y el recordatorio de estas rutinas también se pueden diseñar mediante recordatorios en el calendario del correo profesional.

Figura 3.2: Ticket de rutina de parcheo semestral

3.1.1. Credenciales y accesos

Como se comentó al inicio de este documento, la seguridad informática es muy importante a día de hoy, y es posible que muchas empresas tengan sistemas de monitorización y de vigilancia para evitar accesos no deseados a sus servidores. Por eso es de alta relevancia tratar a través de qué forma se van a acceder a sus servidores y qué creden-

[8] https://www.manageengine.com/products/service-desk/.

ciales se van a usar, puesto que se ha de saber si se va a necesitar la creación de usuarios nominales por temas de seguridad propios de la empresa del cliente o mediante algún usuario genérico que se nos facilite. A continuación, se van a explicar varias formas de acceso.

- **Gestión de conexiones remotas**
  Es la forma más frecuente de realizar los accesos. Hablando desde el marco empresarial, para una empresa de consultoría al parcheo de servidores de múltiples clientes es necesario contar con una herramienta basada en la centralización para poder gestionar y administrar conexiones remotas.

  Estas aplicaciones disponen de ciertas características que facilitarán el trabajo, ya que se puede acceder a varios servidores a la vez, se pueden organizar las máquinas según sea conveniente, además de servir como gestores de credenciales y contar con registros para tener constancia con qué usuarios se ha accedido y en que momentos en caso de que hubiera alguna brecha de seguridad o malas prácticas quedaría constatado en el registro de actividad. Un ejemplo es **Remote Desktop Manager** (RDM)[9].

- **Herramientas externas de seguridad**
  Este apartado se refiere a la obligación de usar herramientas externas para mayor seguridad debido a restricciones impuestas por el cliente para cumplir sus normas de seguridad.

  Este factor extra a tener en cuenta puede requerirse de distintas formas, como la creación de usuarios nominales en plataformas del cliente, el uso de *Multiple-Factor Authentication* (MFA), agregando así una capa extra de seguridad al inicio de sesión usando contraseñas de un solo uso generadas automáticamente cada cierto tiempo o el acceso a través de plataformas monitorizadas para garantizar la seguridad.

  Por ejemplo, puede ser necesario el uso de una Red Privada Virtual o VPN[10] del cliente para poder acceder a su red interna y tener acceso a sus servidores, teniendo de esta forma una capa de seguridad adicional.

  En resumen, puede haber infinidad de casos concretos que se requieran dependiendo del cliente, por eso, previo al parcheo se deberán comentar todos estos factores para poder cumplimentar todas las medidas de seguridad, tanto a nivel nacional como europeo.

3.1.2. Equipos implicados

Lo ideal para que el parcheo de servidores y la mitigación de vulnerabilidades se realice de la forma más eficiente posible es contar con varios equipos encargados de realizar tareas específicas. Es fundamental contar con un equipo centrado en el parcheo de servidores, para este trabajo le llamaremos equipo de operaciones. También se necesitará otro equipo destinado a la monitorización de activos de forma constante, también llamado equipo *24x7*. Para mayor seguridad, sería fundamental contar con un equipo de ciberseguridad, destinado al análisis continuo de vulnerabilidades y análisis de parches para encontrar posibles problemas en ellos. Y por último un equipo de soporte de nivel 2 (llamado N2) encargado de la resolución de problemas que no puedan llevar a cabo los otros equipos.

[9] https://devolutions.net/remote-desktop-manager/.
[10] Del inglés: *Virtual Private Network*.

3.1.3. Grupos de mantenimiento

Un grupo de mantenimiento es considerado un conjunto de activos, en este caso servidores, agrupados para realizar el parcheo de la forma más eficiente posible, debido a tener unas características similares, incluyendo parches a necesitar, herramientas de automatización, ventanas de mantenimiento o forma de actuar en caso de emergencia. Al separar el parcheo en grupos se consigue repartir la carga de trabajo logrando una mayor eficiencia. Además, con la creación de grupos de mantenimiento, se puede crear un grupo de pruebas con servidores de choque para probar el funcionamiento correcto de los parches antes de aplicarlo en entornos de producción.

La única excepción respecto a cuando queramos parchear servidores de las mismas características son con servidores que disponen de alguna función compartida con otros, como pueden ser los controladores de dominio o servidores compartiendo recursos en un *cluster*, debido a su criticidad y la necesidad de realizar un planteamiento específico.

3.1.4. Ventanas de mantenimiento

La ventana de mantenimiento es el margen de maniobra que se tiene para poder realizar el parcheo. En esta ventana se tiene que incluir también tiempo suficiente por si hubiera problemas y se tuviera que realizar el plan de emergencia o de marcha atrás, previamente avisado y acordado con el cliente.

El cliente será el encargado de ofrecernos una ventana de indisponibilidad para cada grupo de mantenimiento que se ha creado en el punto 3.1.3 en función de la criticidad y disponibilidad de estos, ya que para que la actualización surta efecto, en la gran mayoría de casos se requiere un reinicio, provocando un corte en el servidor sobre el que estemos actuando, parando los servicios que este proporcione a otras máquinas, pudiendo dejarlas inoperativas durante ese lapso de tiempo y directamente relacionado, originando pérdidas económicas a la empresa. Por eso para este punto es muy importante la comunicación con el gestor de la empresa a la que estemos ofreciendo los servicios, puesto que esa persona será la encargada de acordar con el equipo a realizar el parcheo las horas y más tarde de transmitir a sus usuarios y encargados, la ventana de mantenimiento para evitar que estén realizando alguna tarea o trabajo a esa hora y pierdan su progreso debido al reinicio. También es importante que no haya nadie conectado, puesto que esos usuarios, pueden ser un problema a la hora de intentar acceder a algún servidor que solo permita uno o dos accesos simultáneos y, por lo tanto, no dejen realizar la instalación correctamente.

En el marco empresarial, el horario de mayor producción y trabajo de oficina es entre las 08:00 y las 20:00, por lo que si el trabajo se acometiera entre estas horas se podría definir como un trabajo en horario laboral, siendo al coste normal, sin embargo, si la empresa que contrata el servicio requiere que el parcheo se realice de madrugada o más tarde de las 20:00, será en horario fuera de hora y el coste será mayor. Pueden requerir de este servicio, sectores importantes como un hospital o un banco, en caso de estar actualizando **servidores de archivos** que no permitiría entrar a historiales clínicos o carpetas bancarias.

Esto también definirá las herramientas y formas de realizar el parcheo que se comentarán más adelante, puesto que una actualización de madrugada se podrá realizar de forma automática, simplemente monitorizando el servidor y sus alertas en caso de no ser crítico, o que se deba estar supervisando de forma activa si así se requiriera.

3.1.5. Uso de herramientas de automatización

En este punto habrá que preguntar al cliente si usan algunas herramientas para poder automatizar el parcheo o realizarlo de forma manual. Esto también dependerá del volumen de servidores del que dispongan o del presupuesto que quieran destinar a estas tareas.

3.2 Sistemas operativos

La diversidad tecnológica actual es amplia y se dispone de distintos sistemas operativos. Los principales y más extendidos mundialmente son Microsoft Windows, Linux, macOS y Unix. De estos, debido a la extensión y variantes que conllevaría, este TFG se centrará únicamente en el sistema operativo desarrollado por **Microsoft** para servidores y entornos empresariales: **Windows Server**.

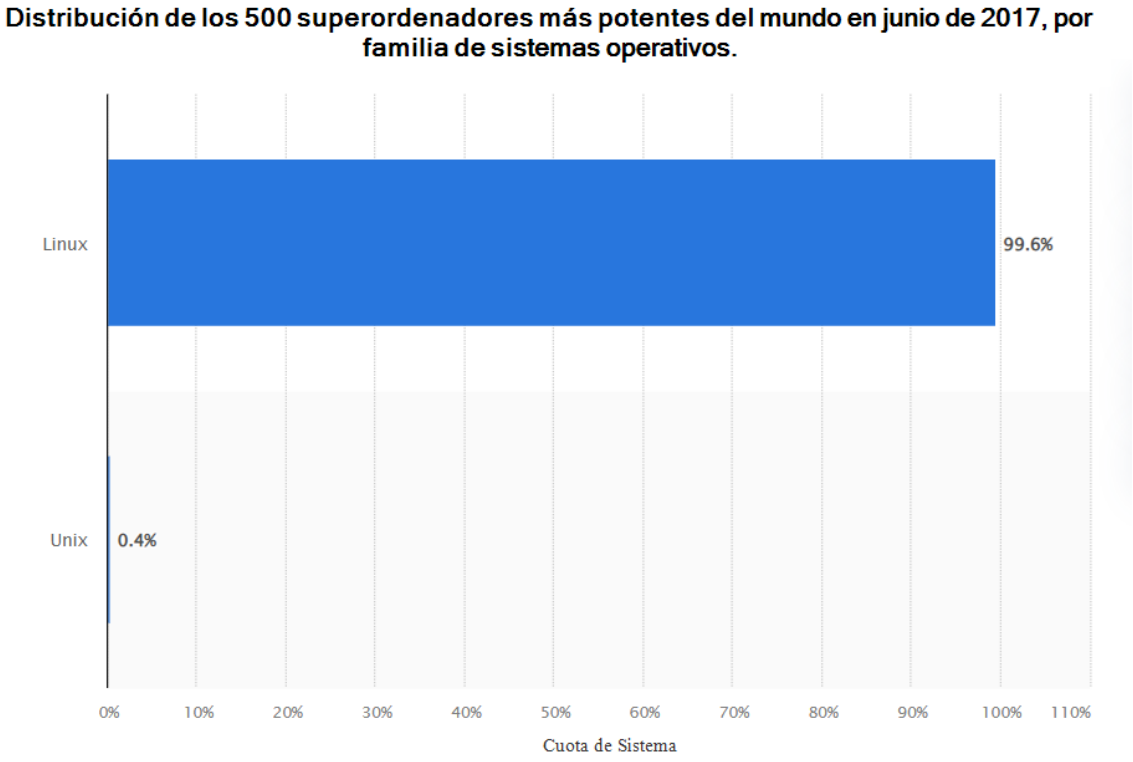


Figura 3.3: Distribución de los 500 superordenadores más potentes del mundo en junio de 2017, por familia de sistemas operativos [11]

Aunque en el entorno de la supercomputación, Linux roza el 100 % de uso gracias a su fácil optimización de recursos y su flexibilidad, como se puede apreciar en la imagen 3.3 donde no llega a aparecer Windows.

En cuanto a sistemas operativos para servidores, Windows dispone de un mayor control del mercado, llegando al 72 % de uso en 2019, gracias a su robustez, fácil aprendizaje y administración, además de la amplia compatibilidad con numerosas empresas, superando ampliamente al resto de sistemas operativos, como se observa en la figura 3.4.

[11] Fuente: **Statista**. Consultar gráfico en: https://www.statista.com/statistics/249270/.

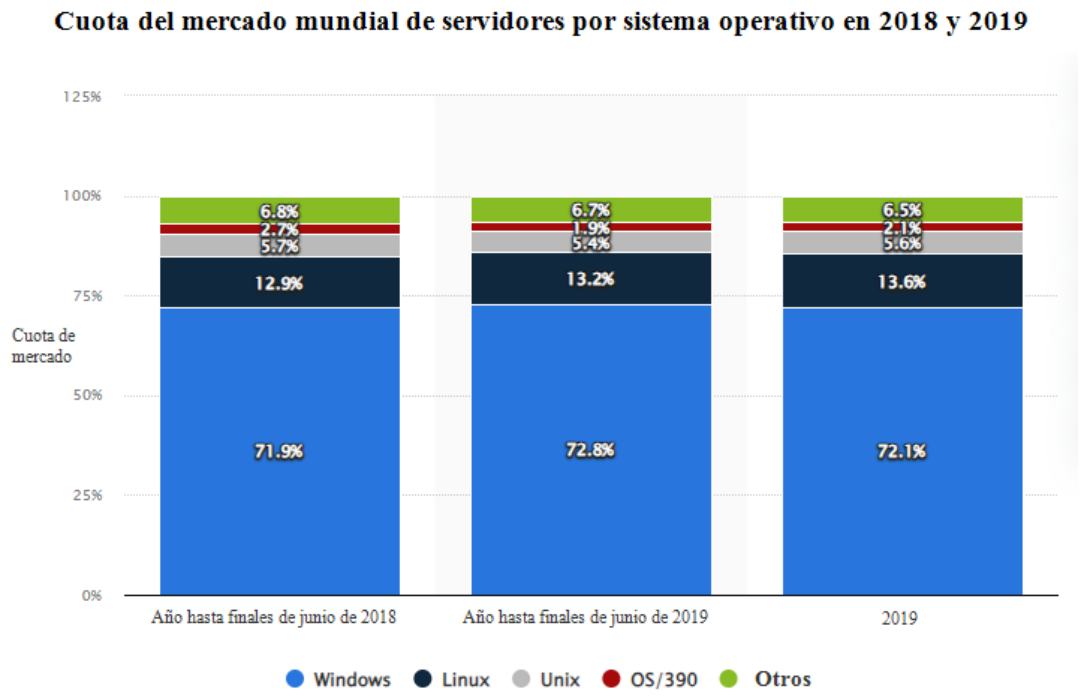


Figura 3.4: Cuota del mercado mundial de servidores por sistema operativo en 2018 y 2019 [12]

En otro estudio de 2018, centrándonos en el ámbito empresarial, vemos como Windows Server sigue siendo líder, rozando el 50 % de uso, mostrado en la figura 3.5.

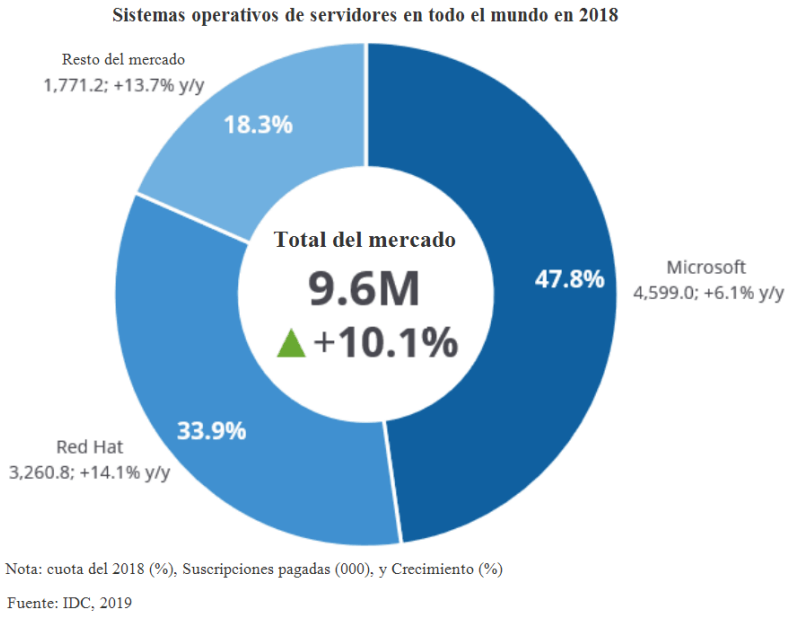


Figura 3.5: Cuota de mercado de sistemas operativos de servidor, 2018 en el ámbito empresarial [13]

[12] Fuente: **Statista**. Consultar gráfico en: https://www.statista.com/statistics/915085/global-server-share-by-os/.

[13] Fuente: **Statista**. Consultar gráfico en: https://www.t4.ai/industry/server-operating-system-market-share.

La elección de realizar este trabajo centrado en el entorno Windows Server, es debido a lo mostrado en las gráficas previas, donde se representaba el amplio uso en el ámbito empresarial comparado con otros sistemas operativos.

3.2.1. Versiones de Windows Server

Con el Windows 2000 surgió la rama empresarial de sistemas operativos con la idea de ofrecer a negocios un mayor rendimiento. Actualmente, se dispone de varias versiones en activo que tienen soporte, de más actual a menos son: 2022, 2019 y 2016. Hasta el 10 de octubre 2023, el sistema operativo Windows Server 2012 R2 contaba con soporte extendido, esto quiere decir que su soporte caducó en 2018, pero **Microsoft** continuaba sacando actualizaciones acumulativas de seguridad para mitigar riesgos de seguridad. Esto último ha ocurrido en mitad de la realización de este trabajo, por lo que a todos los equipos que sean Windows Server 2012 R2, en el mes de octubre de 2023, recibieron su último parcheo y por ello dejaron de entrar en los ciclos de parcheo de los clientes, pudiendo como mucho hacerles un reinicio preventivo, siendo este concepto parte de la estrategia *Software Rejuvenation*, basada en reiniciar periódicamente el servidor para prevenir la acumulación de errores y que no influya en el rendimiento, así consiguiendo mejorar la confiabilidad y el rendimiento del sistema [12]. Para que se pudiera seguir incluyendo en el ciclo de parcheos, se debería migrar a una versión más reciente como las mencionadas anteriormente.

3.2.2. Catálogo de Microsoft Update

Lo primero de todo es fijar la fecha de salida de actualizaciones de seguridad críticas y acumulativas mensuales para poder preparar la planificación. Las nuevas actualizaciones de **Microsoft**, se publican en su catálogo el segundo martes de cada mes por la tarde, comúnmente conocido como *Patch Tuesday*. Estas actualizaciones acumulativas cubren vulnerabilidades que se hayan descubierto recientemente además de proporcionar mejoras de seguridad y de rendimiento. También pueden solucionar problemas existentes o *bugs* de versiones antiguas. En caso de que hubiera alguna amenaza crítica de seguridad, **Microsoft** podría publicar una actualización correctiva fuera de este día.

Para la comprobación de los parches de seguridad se accede a la página del **Catálogo de Microsoft Update**[14]. Desde ahí, aparecerá un cuadro de búsqueda donde se puede introducir el parche que se quiere buscar, el mes o el sistema operativo del servidor que se quiere parchear. Un ejemplo de búsqueda puede ser: *2024-01 Windows Server 2016*. Con esto se busca cualquier actualización referente al mes de enero de 2024 para el sistema operativo Windows Server 2016. De esta forma, aparece como resultado la actualización acumulativa de seguridad para sistemas Windows Server 2016 entre otros resultados como se aprecia en la figura 3.6. En la figura se pueden observar distintas características del parche: el título completo de la actualización, el producto al que se aplica, la clasificación que le otorga **Microsoft**, la fecha de salida del parche y el tamaño que ocupa, además de la opción de descargar el parche. Con esta primera vista general se obtiene bastante información.

En el apartado de **Título** se muestra el nombre que **Microsoft** le otorga al parche, como se ve en la figura 3.6, añadiendo el identificador *KB5034119*. Las letras **KB** significan *Knowledge Base Article* o en castellano "Artículo de Base de Conocimiento". Los números posteriores son un identificador único asignado a un artículo específico en la Base de Conocimiento de **Microsoft**. Con este identificador único se facilita la búsqueda para ver

[14] https://www.catalog.update.microsoft.com.

problemas conocidos o encontrar soluciones específicas para ese parche. Por comodidad, en ciertos momentos, se usará el término *KB* para referirse a los parches proporcionados por **Microsoft**.

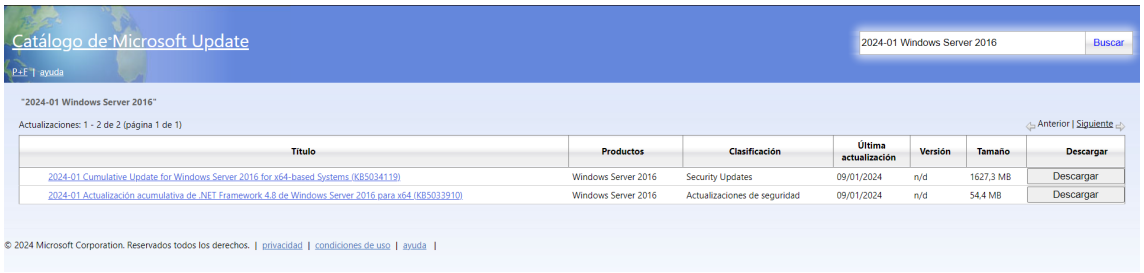


Figura 3.6: Catálogo de Microsoft Update para Windows Server 2016

Si se clica sobre el parche deseado, aparecerá una ventana en la cual se mostrarán diferentes secciones con información relevante sobre el parche seleccionado. En el apartado de **Información general** se muestra una breve descripción del parche y un enlace a la propia página de soporte de **Microsoft** con más información acerca del parche y con posibles problemas relacionados.

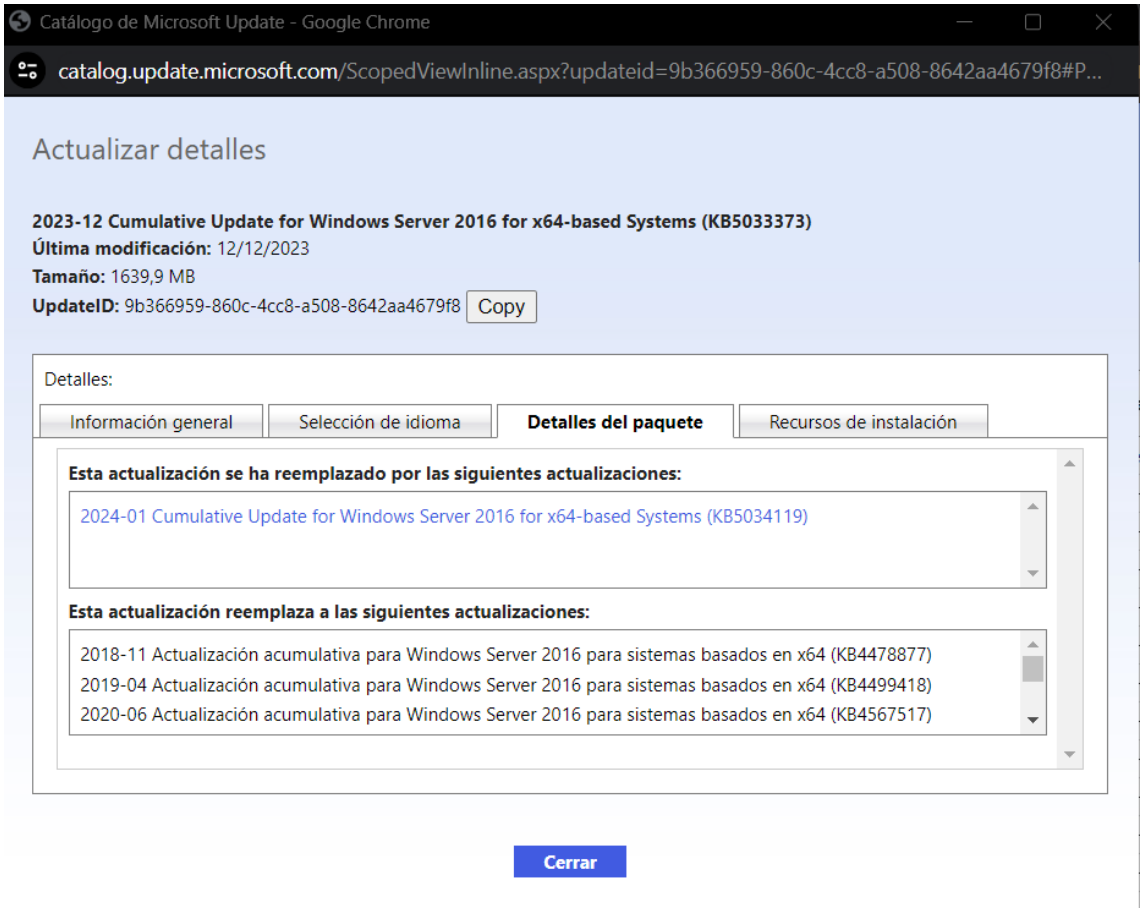


Figura 3.7: Reemplazo de la actualización

Para buscar los parches mensuales publicados, en lugar de filtrar por versión de sistema operativo o por fecha, para evitar que aparezcan varias actualizaciones de distintos productos, se puede identificar el parche del mes anterior al que se quiere reemplazar con su identificador único. Desde el apartado **Detalles del paquete** como se muestra en

la figura 3.7 se ha buscado el parche referente al SO Windows Server 2016 de diciembre de 2023 y se detallan tanto las actualizaciones que reemplazó este parche, como la actualización que va a reemplazarlo. Clicando sobre el nuevo parche él se abrirá y se verá la información referente al nuevo parche.

Todo esto es aconsejable ir apuntándolo y dejándolo guardado en un documento de seguimiento, para tener constancia de todos los parches que han ido saliendo para a la hora de tener que enviar correos con la planificación o buscar si el parche está instalado correctamente tenerlo a mano.

3.2.3. Pilas de servicio y posibles problemas de los parches

Primero se ha de explicar qué es una pila de servicio, también conocida como pila de mantenimiento. Es un elemento clave para la instalación de las actualizaciones de Windows, además de ser importante para la implementación de Windows, características y roles. A diferencia de las actualizaciones acumulativas mencionadas anteriormente, las pilas de servicio no se publican mensualmente, sino solo cuando ocurren problemas nuevos por nuevos parches de seguridad o cuando surgen vulnerabilidades. Son de gran importancia, puesto que si no están ciertas pilas de servicio instaladas en el equipo puede hacer que la máquina que se pretenda subir de versión dé errores en la instalación o no se pueda actualizar [13].

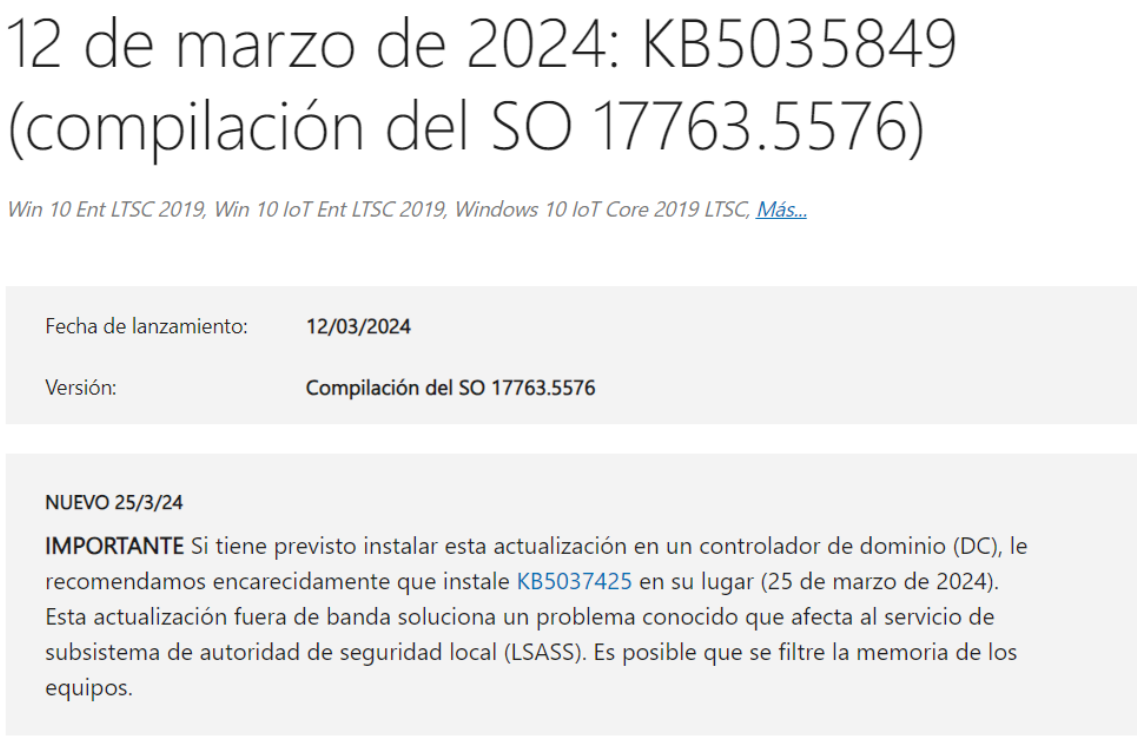


Figura 3.8: Problemas de filtrado de memoria en el parche de marzo

Se debe comprobar si existen problemas conocidos desde el enlace que mencionó en el apartado 3.2.2 que redirige a la página de soporte de **Microsoft** para ver más información sobre la actualización. Está el apartado **Problemas conocidos en esta actualización** donde se explican que inconvenientes tiene este parche dando una información precisa de por qué ocurre y si existen posibles soluciones. Es importante investigar y comprobar si estos problemas pueden ser relevantes a la hora de realizar el parcheo, en algunos casos el cliente debe consultar con el proveedor del servicio que está ejecutándose para

confirmar si el parche a aplicar tiene afectación sobre este servicio. En caso afirmativo, quizá se deba no aplicar ese parche o parar algunos servicios o aplicaciones previamente.

Como se puede observar en la figura 3.8 durante el mes de marzo de 2024, para todas las versiones de Windows Server con soporte, el parche mensual que salió en el *Patch Tuesday*, llevaba un error de filtrado de memoria de los equipos, afectando directamente a los controladores de dominio. Un *Memory leak* se refiere a la acumulación gradual de memoria por parte de las aplicaciones, siendo el sistema incapaz de liberar esa memoria, lo que puede resultar en fallos, reinicios inesperados y un consumo excesivo de recursos del sistema. **Microsoft** alertó de este problema diez días después de la salida del parche, y tres días más tarde sacó un parche de emergencia para solucionar estos problemas. Aquí reside la importancia de revisar los problemas conocidos, documentarse correctamente, dar un margen de tiempo para evitar las vulnerabilidades *Zero-day*, y tener un grupo de choque para probar primero los parches. Incluso, pudiendo llegar a ser recomendable aplicar los parches acumulativos de seguridad con un mes de diferencia, para evitar problemas que se hayan podido ir documentando durante todo el mes y prevenirlos. Esta última práctica no es aconsejable si el parche del mes actual soluciona alguna vulnerabilidad crítica para la organización.

Es recomendable juntar toda la información necesaria en el documento de seguimiento mencionado anteriormente, para tenerlo todo de una manera visible y organizada y a la hora de tener que elaborar alguna planificación, tomar este documento como premisa.

Septiembre				
KB	Fecha Publicacion KB	Version SO	Posibles Pilas de Servicio	Problemas Conocidos
KB5030278	12/09/2023	Windows Server 2012	Pila KB5030330	
KB5030278	12/09/2023	Windows Server 2012 R2	Pila KB5030329	
KB5030213	12/09/2023	Windows Server 2016	Pila KB5030504	
KB5030214	12/09/2023	Windows Server 2019	Pila KB5005112	
KB5030216	12/09/2023	Windows Server 2022	-	Desactivar Secure Boot y Problemas inicio con ESX 7.0 o anterior
KB5030211	12/09/2023	Windows 10 21H2 y 22H2	-	

Octubre				
KB	Fecha Publicacion KB	Version SO	Posibles Pilas de Servicio	Problemas Conocidos
KB5031442	10/10/2023	Windows Server 2012	Pila KB5031469	
KB5031419	10/10/2023	Windows Server 2012 R2	Pila KB5030329	
KB5031362	10/10/2023	Windows Server 2016	Pila KB5031467	
KB5031361	10/10/2023	Windows Server 2019	Pila KB5005112	
KB5031364	10/10/2023	Windows Server 2022 21H1		Desactivar Secure Boot
KB5031356	10/10/2023	Windows 10 21H2 y 22H2	Pila KB5011543	

Noviembre				
KB	Fecha Publicacion KB	Version SO	Posibles Pilas de Servicio	Problemas Conocidos
Out of support	Se instala la de octubre	Windows Server 2012	-	
Out of support	Se instala la de octubre	Windows Server 2012 R2	-	
KB5032197	14/11/2023	Windows Server 2016	Pila KB5032391	
KB5032196	14/11/2023	Windows Server 2019	Pila KB5005112	Problema visual cifrado disco
KB5032198	14/11/2023	Windows Server 2022 21H1	-	
KB5032189	14/11/2023	Windows 10 21H2 y 22H2	Pila KB5003173 - Pila KB5005260	Problema visual cifrado disco

Figura 3.9: Documento de seguimiento de los parches

Fijándonos en la imagen 3.9 se observan las distintas actualizaciones acumulativas de seguridad, con sus pilas de servicio necesarias y con posibles problemas que tienen, como problemas visuales de cifrado de disco, que simplemente es un *bug* visual, pero que si no se notifica al cliente de que les puede saltar ese error gráfico se podrían pensar que hay algún problema serio con sus discos cifrados. También se aprecian otros errores, como tener que desactivar el *Secure Boot*, siendo este un arranque seguro de Windows. También se ve en el propio documento de seguimiento como a partir de noviembre las versiones de Windows Server 2012 y 2012 R2 dejaron de estar en soporte.

3.3 Servidores necesarios en una organización

En este apartado se van a detallar distintos tipos de servidores que son necesarios en una organización para su correcto funcionamiento.

3.3.1. Controladores de dominio

Un **controlador de dominio** o DC[15] constituye un elemento vital en la estructura del **Directorio Activo** o AD[16], desplegando un rol esencial en la administración y unificación de la autenticación de usuarios, además del control de acceso a los recursos de red dentro de un dominio determinado. Esencialmente, opera como el núcleo central en un entorno de dominio Windows, supervisando los privilegios de acceso a distintas áreas de la red y verificando la identidad de los usuarios. El AD es el servicio de directorio de **Microsoft** utilizado para almacenar información sobre recursos de red, incluidos usuarios, grupos, impresoras y otros dispositivos. Está basado en el estándar **Protocolo Ligero de Acceso a Directorios** (LDAP[17]) y se organiza en una estructura jerárquica. El AD proporciona una serie de características, como la replicación de datos entre controladores de dominio, la integración con servicios de seguridad como **Kerberos**, y la capacidad de administrar directivas de grupo o GPO[18] [17][18].

Las directivas de grupo son herramientas de administración en entornos Windows que permiten definir configuraciones y restricciones para usuarios y computadoras dentro de un dominio del AD. En el contexto del parcheo de servidores, las directivas de grupo se pueden utilizar para definir políticas de actualización de Windows que especifiquen cuándo y cómo se aplicarán las actualizaciones de seguridad en los servidores de la red. Por ejemplo, se pueden configurar políticas para descargar e instalar automáticamente las actualizaciones de seguridad en determinados horarios, o para permitir a los administradores programar y controlar manualmente el proceso de parcheo [19].

En entornos empresariales, salvo para entornos pequeños y entornos de prueba, lo lógico es contar con más de un DC, para así garantizar disponibilidad, balanceo de carga y tolerancia a fallos, entre otras características, además de ofrecer una facilidad de mantenimiento, gracias a la redundancia, dado que se puede parchear un DC sin interrumpir los servicios para los usuarios.

En resumen, los controladores de dominio y el directorio activo son componentes fundamentales en entornos de red Windows, mientras que las políticas de grupo proporcionan herramientas poderosas para administrar la configuración y seguridad de los sistemas, incluido el parcheo de servidores.

3.3.2. Servidor de backup

Con este servidor, se van a poder hacer copias de datos o archivos, para poder restaurarlos en caso de que haya una pérdida de datos o fallos en los discos. Las copias de seguridad se almacenan en un soporte diferente al del almacenamiento principal y la velocidad de una copia de seguridad depende de la velocidad de transferencia y de la cantidad de archivos o carpetas que la tarea tenga asignados.

[15] Del inglés *Domain Controller*.
[16] Del inglés *Active Directory*.
[17] Del inglés, *Lightweight Directory Access Protocol*.
[18] Del inglés *Group Policy Object*.

Para gestionar los *backups* de los servidores de la organización, existen distintas empresas centradas en el *software de gestión de backup*. Una de ellas es **Veeam**[19]. Para gestionarlos desde una forma centralizada se requiere contar con la infraestructura de *Veeam Backup & Replication*, desde donde se administrarán todos los *backups*, consiguiendo una administración central, configurando una política de copias de seguridad según se desee. Otra forma de poder administrar los *backups* de una forma independiente o *standalone*, para servidores físicos o que se administran de forma individual, es mediante la instalación del agente de *backup* en el propio servidor, pudiendo gestionarse de forma independiente con cualquier usuario con permisos de administrador en el servidor y así pudiendo configurar una política de copias de seguridad según se desee. Un ejemplo es la imagen 3.10, donde se ve el agente de *backup* de un servidor, gestionado de forma *standalone*, viendo en la imagen seis *backups* incrementales y uno completo. En lugar de que el *backup* se gestione desde el propio servidor de forma independiente mediante el agente, se puede gestionar desde la infraestructura de *Veeam Backup & Replication*, desde donde se administrará el agente de *backup* y su configuración, siendo esta forma interesante para servidores con una administración central [20].

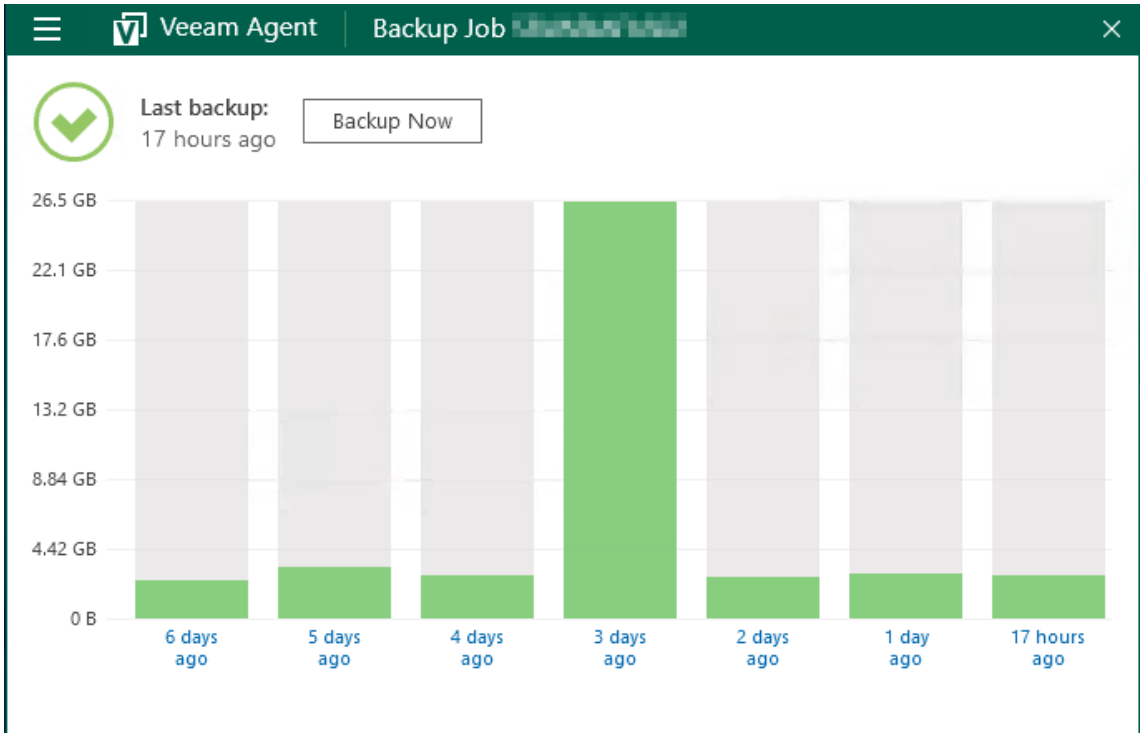


Figura 3.10: Agente de *backup* de **Veeam** gestionado por el propio servidor

Estos *backups* pueden ser copias de seguridad completas, también llamadas *full backups* o pueden ser *backups* incrementales o diferenciales. Estos últimos son *backups* que solo copian archivos de nueva creación o que han sufrido modificaciones desde el *backup* anterior, consiguiendo optimizar el espacio de almacenamiento. Para una combinación óptima lo ideal sería realizar *full backups* con un margen de tiempo y acompañarlos de *backups* incrementales. Por ejemplo, realizar un *full backup* todos los sábados de madrugada y que el resto de días se realicen *backups* incrementales. En la figura 3.10 se puede observar una configuración similar.

En la imagen 3.11 se puede observar el resumen de la creación de un *backup* para un servidor físico *SRV-TECDC01*. En este caso se ha configurado para que el *backup* sea

[19] https://www.veeam.com/es.

controlado por el servidor de *backup* y se haga *backup* de la máquina entera. También se puede comprobar la *Retention Policy* o puntos de retención que se han configurado en catorce días. Esto quiere decir que un *backup* estará almacenado durante catorce días por seguridad en caso de tener que recuperar algún archivo o carpeta de algún día en específico.

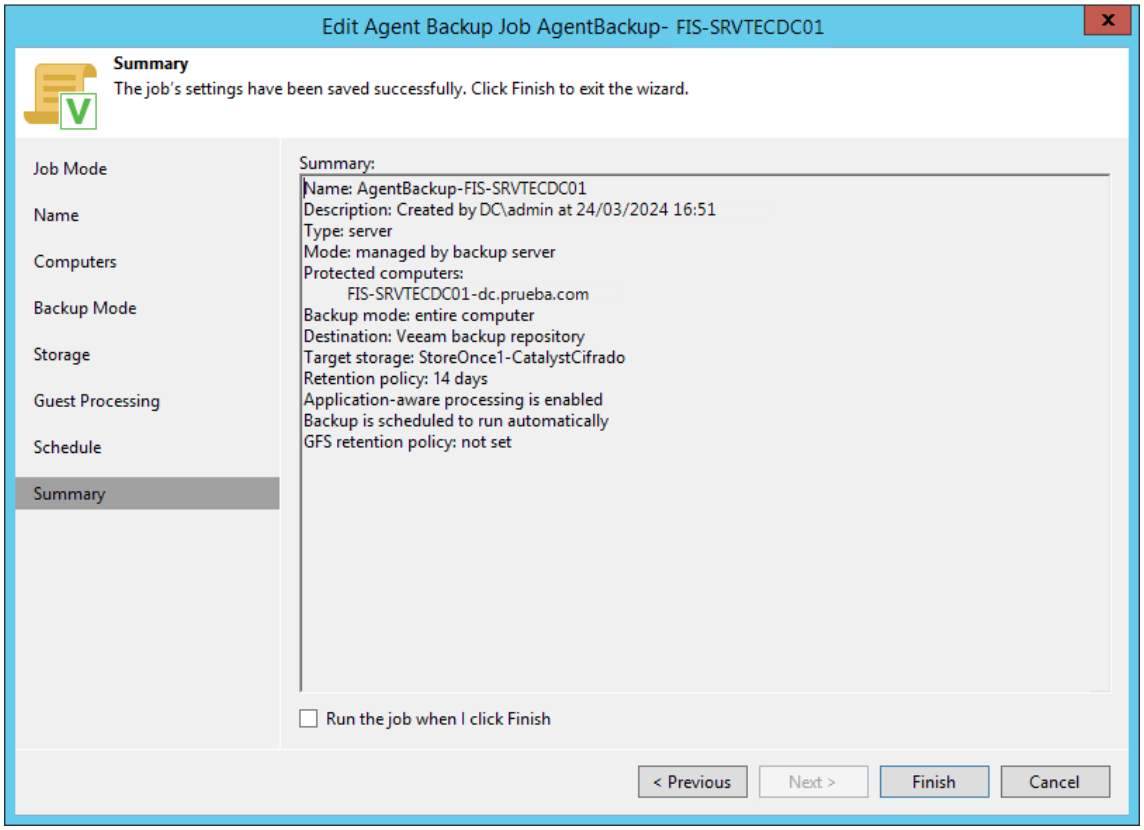


Figura 3.11: Resumen de la creación de un *backup* para un servidor físico desde **Veeam**

De forma periódica y especialmente cuando se va a parchear el servidor, se debe revisar que los *backups* se realizan correctamente, por si fallara el parcheo, y hubiera que restaurar el servidor desde la última copia de seguridad disponible.

3.3.3. Servidor de bases de datos

Otro servidor importante en una organización es un servidor de bases de datos. Este servidor es necesario para la gestión y administración de forma centralizada de las bases de datos, permitiendo así el acceso a la información de manera eficiente y segura. Con este servidor se logra almacenar, organizar y mantener grandes cantidades de información de una forma eficiente. Son de gran importancia estos servidores, ya que otras aplicaciones y sistemas dentro o fuera de la organización pueden depender de ellos.

Usando un servidor de bases de datos, se logra reducir la probabilidad de perder información relevante en la empresa, ya que se usa un acceso centralizado a los datos, además de un acceso restringido, permitiendo únicamente a ciertos usuarios permisos de escritura. Además, se deben usar copias de seguridad para lograr tener un sistema de respaldo en caso de que hubiera que realizar alguna restauración de datos.

3.4 Diversidad tecnológica

En la estructura tan dinámica de la informática actual, la gestión efectiva de los servidores es un gran desafío, por lo que en este punto se abarcarán distintas formas de acceso y parcheo de los servidores.

3.4.1. Hipervisores

Un hipervisor o supervisor de máquina virtual, es un software que permite crear y ejecutar **máquinas virtuales** (MV). Además, aisla el sistema operativo y los recursos de las máquinas virtuales, permitiendo su creación y gestión. En resumen, los hipervisores son esenciales para la virtualización y han revolucionado la forma en que se utilizan y gestionan los recursos informáticos [14]. Existen 2 tipos de hipervisores:

- Los hipervisores **Bare-Metal**. Estos se ejecutan directamente sobre el hardware físico, sin necesidad de un sistema operativo anfitrión intermedio, permitiendo un mayor rendimiento, ya que no hay una capa adicional entre hipervisor y hardware, además ofreciendo una mayor eficiencia y control directo sobre los recursos físicos, siendo ideales para centros de datos y entornos empresariales, por esto último serán los que se expongan en este trabajo.
- Los hipervisores **alojados** ejecutándose como aplicaciones dentro de un sistema operativo anfitrión, siendo más sencillos de instalar y configurar, además proporcionar una mayor flexibilidad y conveniencia para usuarios individuales o pequeñas empresas [15]. Un gran ejemplo es **VirtualBox**[20] siendo un software de código abierto creado por Oracle, muy útilmente usado durante la carrera.

La figura 3.12 muestra datos de un estudio realizado por *SpiceWorks* analizando más de 530 empresas de TI tanto de Europa como de América sobre el estado de la tecnología de virtualización en 2020, comparando distintos tamaños de empresas, el uso de herramientas y las preferencias que tienen acerca de estas herramientas [16].

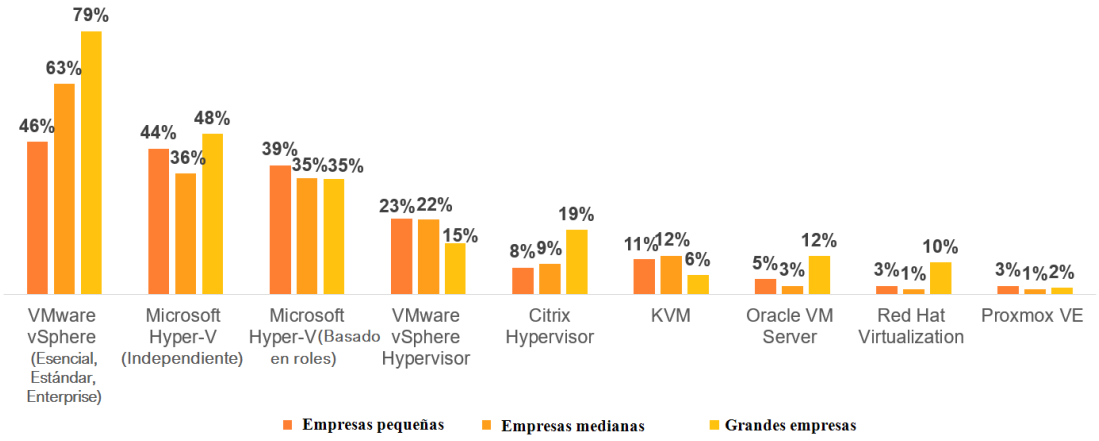


Figura 3.12: Comparación de uso de hipervisores en 2020 [16]

[20] https://www.virtualbox.org/.

Respecto a esos hipervisores podemos destacar **VMware ESXi**[21], **Microsoft Hyper-V**[22], **KVM**[23] y **Citrix Hypervisor**[24]. En la tabla 3.1 se detallan en profundidad algunas características y requisitos de esos hipervisores.

Tabla 3.1: Comparación hipervisores Tipo 1

Hipervisor	Características	Capacidades	Requisitos
VMware ESXi	Hipervisor bare-metal ampliamente utilizado en centros de datos.	Soporte completo para virtualización de servidores y estaciones de trabajo. Alta disponibilidad y recuperación ante desastres. Gestión centralizada con vSphere.	Plataforma de servidores compatible. CPU con al menos dos núcleos. Bit NX/XD habilitado en la CPU. Mínimo de 4 GB de RAM física. Controladoras Gigabit o Ethernet. Disco de arranque de al menos 32 GB. Disco SCSI o LUN RAID local.
Hyper-V	Hipervisor de Microsoft adecuado para entornos empresariales.	Integración con Windows Server y System Center. Soporte para contenedores y microservicios. Live Migration y Replica para continuidad del negocio.	Procesador de 64 bits con SLAT. CPU compatible con VT-x (Intel). Mínimo de 4 GB de memoria.
KVM	Hipervisor bare-metal de código abierto con buen rendimiento.	Virtualización basada en Linux con escalabilidad y seguridad. Soporte para QEMU y libvirt. Migración en vivo y gestión de recursos.	Hardware compatible con virtualización. CPU con extensiones VT-x o AMD-V.
Citrix Hypervisor	Anteriormente conocido como XenServer, es una opción comercial popular.	Optimización para VDI y aplicaciones críticas. Integración con Citrix Virtual Apps and Desktops. Funciones avanzadas de red y almacenamiento.	Al menos dos equipos físicos x86 independientes. Un servidor para Citrix Hypervisor y otro para XenCenter o la interfaz de línea de comandos.

Además, los hipervisores tienen la ventaja de que se pueden agrupar, esto implica agrupar varios servidores físicos para formar un *cluster*. Esto ofrece beneficios como alta disponibilidad, balanceo de carga y migración de máquinas virtuales entre nodos. Este proceso se detallará más adelante cuando se explique la ejecución del parcheo para un nodo del hipervisor.

[21] https://www.vmware.com/products/esxi-and-esx.html.

[22] https://learn.microsoft.com/es-es/windows-server/virtualization/hyper-v/hyper-v-on-windows-server.

[23] https://www.linux-kvm.org/page/Main_Page.

[24] https://www.citrix.com/platform/citrix-hypervisor/.

3.4.2. Infraestructura en la nube

También existen plataformas en entornos de nube, para gestionar infraestructuras que permiten gestionar múltiples servidores de forma centralizada, gestionando los servidores desde una única interfaz o mediante una *Application Programming Interface* (API)[25] proporcionada por el proveedor de la nube. El proveedor de servicios aloja y gestiona el hardware físico en sus centros de datos, mientras que los clientes pueden crear y configurar las máquinas virtuales para ejecutar aplicaciones, servicios o cargas de trabajo, pudiendo aumentar o disminuir la capacidad, cambiar la configuración, consiguiendo así escalabilidad y flexibilidad, además de permitir un acceso global.

Además, permiten la automatización de operaciones, como apagados o encendidos automáticos, aplicación directa de parches de seguridad, la monitorización del rendimiento y recuperación ante desastres. Uno de los mayores ejemplos de plataformas de gestión en la nube es **Microsoft Azure**[26].

3.4.3. Servidores físicos

A diferencia de un servidor virtual, que opera dentro de un entorno virtualizado, un servidor físico es una máquina independiente y autónoma. Parchear un servidor físico requiere consideraciones específicas relacionadas con la disponibilidad del servicio, la compatibilidad del hardware, la gestión de recursos y la planificación cuidadosa de las pruebas y copias de seguridad. Actualmente, los servidores físicos pueden ser administrados de forma remota a través de conexiones de red, lo que permite monitorizarlos, configurarlos y mantenerlos sin necesidad de estar físicamente presentes en el lugar donde está ubicado el servidor.

3.5 Plan de emergencia

En este apartado se fijarán una serie de pasos en caso de que ocurran problemas durante la instalación o a posteriori. Esto se va a diferenciar con la ventana de reinicio establecida previamente. El primer paso es establecer con el cliente y los equipos involucrados, una manera de poder dar marcha atrás al parcheo para volver a un estado donde el servidor funcionaba correctamente.

Para unas buenas prácticas, se debe contar con copias de seguridad periódicas de los servidores, para garantizar la seguridad de todos los datos y evitar la pérdida ante posibles problemas.

El primer paso de marcha atrás común en todos los servidores en caso de que el parche haya dado problemas o no se haya instalado correctamente será revisar el error, herramientas como el **Visor de eventos** de Windows, donde se puede encontrar más información de errores o alertas y buscar esos errores posteriormente para intentar resolver el problema. En caso de no poder solucionar el fallo, el siguiente paso es intentar desinstalar el parche. Desde el panel de control del servidor se irá a **Programas**, clicaremos en **Ver actualizaciones instaladas** y procederemos a desinstalar la causante del problema. Si no es posible, habrá que arrancar el servidor en modo seguro, intentando desinstalar la KB desde ahí. En caso de que no sea posible o continúen los problemas, a continuación, se van a reflejar posibles planes de marcha atrás según el tipo de servidor que se esté parcheando.

[25] Interfaz de Programación de Aplicaciones.
[26] https://azure.microsoft.com/es-es.

3.5.1. Máquinas virtuales

Principalmente, se conseguirá mediante la realización de una instantánea también llamado punto de control o *snapshot*. Una *snapshot* es un método para volver a la configuración existente en el momento en que esta se creó. Una *snapshot* no copia datos, sirve principalmente como protección ante cambios de configuraciones o actualizaciones, siendo su contenido guardado en el mismo volumen que la máquina virtual esté almacenada. En caso de que hubiera una degradación de discos o fallos del sistema de almacenamiento principal, al estar en el mismo volumen se perdería la *snapshot*, mientras que una copia de seguridad no. Una *snapshot*, como su propio nombre indica, es instantáneo, por lo que se realiza en muy poco tiempo [21][22].

Esto no quiere decir que haya que sustituir las copias de seguridad periódicas por *snapshots*, ya que otro aspecto significativo que demuestra que una *snapshot* no es una alternativa adecuada a las copias de seguridad radica en su incapacidad para recuperar elementos de manera selectiva, como un archivo o carpeta. Aunque una *snapshot* pueda restaurar una máquina virtual a un estado anterior completo, carece de la capacidad para recuperar archivos o aplicaciones específicas dentro de esa máquina virtual. Además, muchos servidores de aplicaciones están interconectados con otros servidores, lo que significa que una aplicación puede depender de un servidor SQL, una interfaz web o un servidor LDAP, entre otros. Al utilizar un punto de control para revertir un servidor de aplicaciones a un estado anterior, existe el riesgo de generar inconsistencias, ya que los otros servidores de los que depende no se someten al mismo proceso de reversión. Si bien el grado de riesgo puede variar según la aplicación y el papel de la máquina virtual, es fundamental considerar siempre la coherencia de la aplicación al recurrir a *snapshots* [22]. En estos casos puede ser necesario recurrir a un *backup* en lugar de a una *snapshot*. Como en el caso de un servidor de bases de datos, al revertir una *snapshot*, si no ocurre una sincronización correcta entre la base de datos y el servidor puede haber inconsistencias. También puede darse el caso que se pierda alguna transacción que se estuviera realizando en el momento de la *snapshot*, provocando datos incompletos, erróneos o que haya problemas de conexión debido a dependencias de otros sistemas a este servidor. Por todo esto, se debe realizar un buen análisis previo al parcheo de cada servidor para establecer su plan de emergencia de forma correcta y precisa.

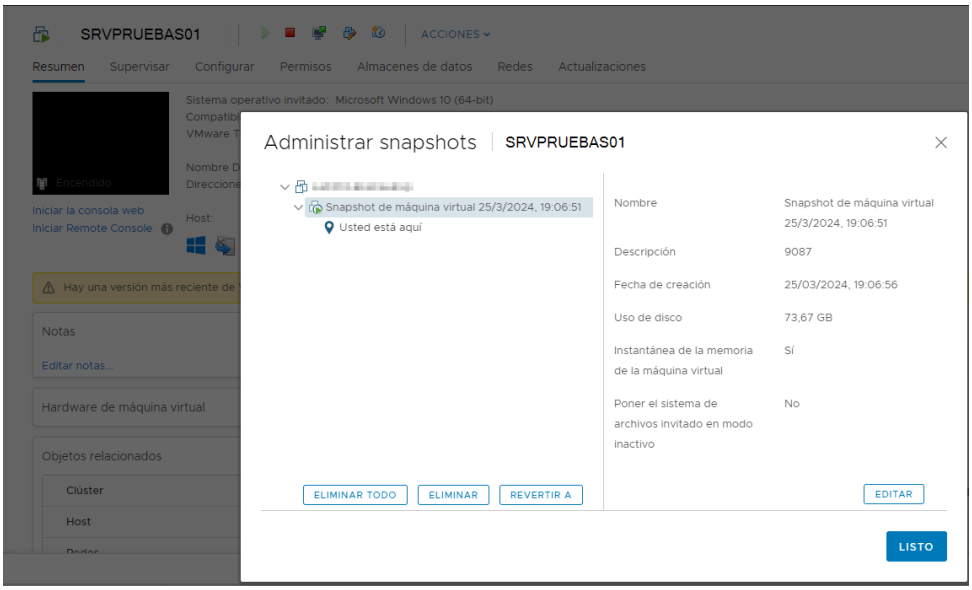


Figura 3.13: *Snapshot* de una máquina virtual desde **VMware vCenter**

Las *snapshots* se realizarán desde distintos programas de virtualización, como **VMware vCenter**[27], como se observa en la figura 3.13. Es muy importante tener en cuenta que las *snapshots* son incrementales, esto quiere decir que conforme pase el tiempo la imagen irá aumentando en tamaño, hasta poder consumir el espacio de almacenamiento en su totalidad y afectar gravemente al rendimiento del sistema, no recomendándose que esté activa más de 72 horas. Por eso, cuando se va a realizar la intervención, la *snapshot* se crea horas antes de realizarlo y se recomienda su eliminación al día siguiente de la actualización, cuando se ha comprobado el correcto funcionamiento del sistema, cuando el cliente o el resto de equipos implicados den el visto bueno para poder borrarla [23].

Puede darse el caso de que posterior al reinicio requerido por la actualización, se haya instalado la *KB* con problemas. Lo primero de todo habría que revisar si seguimos dentro del tiempo establecido, en caso de ser así, se intentaría volver a instalar la actualización y si el fallo persistiera se revertiría la *snapshot* para volver al estado anterior. Tras haber mitigado el problema se contactaría con el cliente para comentar la situación e intentar replanificar el parcheo a otra fecha. También se ha de comentar que tras finalizar el parcheo, se debe exigir al cliente probar sus servidores, debido a que el parche puede ser que interfiera con aplicaciones de terceros, siendo esto algo que por nuestra parte no podemos comprobar.

3.5.2. Servidor físico

En caso de parchear un servidor físico, no se puede recurrir a una *snapshot*, puesto que estos servidores no se encuentran en plataformas de virtualización como sí lo están las máquinas virtuales mencionadas en el punto anterior. Para tener un plan de marcha atrás correcto para un servidor físico, se deben realizar copias de una forma periódica, como se explicó en el apartado 3.3.2, en la aplicación de *backups*, para que en caso de que fallara la instalación, se hiciera una restauración de *backup* del día deseado.

Antes de intentar una restauración de *backup* en algunos servidores físicos, existe una interfaz de administración remota llamada ILO[28] o *Integrated Ligths-Out*, que como el propio nombre indica, permite controlar y gestionar el servidor aunque esté apagado o inaccesible físicamente. La ILO permite acceso al servidor mediante una consola remota, incluso cuando el servidor físico no es accesible de forma remota debido a algún problema o que haya arrancado en modo seguro. También se le puede provocar un reinicio remoto o controlar actualizaciones. Esta herramienta suele ser frecuente en marcas como **HP** y **Dell**.

3.5.3. Controladores de Dominio

Respecto a los controladores de dominio, como se ha comentado en el punto anterior, los DC deben estar redundados para en caso de que uno de los controladores de dominio experimente fallos, los otros puedan asumir la carga sin tener gran impacto en la infraestructura. Vamos a diferenciar entre virtualizados y físicos, en caso de que se tengan dos o más DC redundados o con la base de datos en un disco independiente:

- **DC virtualizado:** En caso de no haber podido desinstalar el parche y que el DC esté inoperable, una respuesta rápida ante el problema sería lo siguiente: Eliminar el DC en cuestión, desconectándolo de la red para evitar conflictos, además de eliminar la cuenta de máquina del DC que ha fallado. También si dispone de los roles

[27] https://www.vmware.com/products/vcenter.html.
[28] https://www.hpe.com/es/es/what-is/ilo.html.

FSMO[29] se deben transferir mediante el comando *ntdsutil* desde Powershell a otro DC. Posteriormente, hay que reemplazar el controlador fallido instalando un nuevo controlador. Esto se puede realizar promocionando un nuevo servidor a controlador de dominio, para sustituir el fallido.

Otro método, si se dispone de una copia de seguridad del estado del sistema, se puede restaurar el DC, iniciándolo en el modo de restauración de servicios de directorio (DSRM[30]). Para ejecutar este proceso (comúnmente llamado *Disaster Recovery*), se debe seguir con las mejores prácticas indicadas por **Microsoft** [24]. Posteriormente, se debe verificar que los registros de **sistema de nombres de dominio** o DNS[31] se actualizan de forma correcta y comprobar que funciona la replicación entre los controladores. Tal como se indica anteriormente, es importante arrancar la copia en el modo DSRM para evitar, entre otros problemas, conflictos de **Número de Secuencias Actualizadas** o USN[32]. El USN es un valor asignado a cada modificación realizada en un objeto dentro de la base de datos del AD. Cada controlador de dominio mantiene su propio USN para registrar las actualizaciones efectuadas en su copia local de la base de datos. Pueden surgir ciertos problemas si los controladores de dominio no sincronizan adecuadamente sus USN debido a fallos de replicación o configuraciones incorrectas, pudiendo provocar un *USN Rollback* o reversión de USN, desembocando en que deje de recibir actualizaciones y causando problemas en todo el AD. Para evitar esto, se restaura el DC virtualizado desde una copia de seguridad muy reciente [25].

- **DC físico:** Habría que seguir el mismo plan de recuperación que el explicado para un controlador de dominio virtualizado, salvo por la diferencia de si no sirve la restauración de la copia de seguridad, habría que reconstruir un servidor físico.

En caso de que en la organización exista un único controlador de dominio, la única opción es recuperar de la copia de seguridad, teniendo en cuenta el impás de tiempo entre el tiempo de caída y la copia, durante esa franja de tiempo cualquier cambio realizado se perderá y habrá que volver a realizarlo.

Los controladores de dominio desempeñan un papel crucial en la infraestructura de red. En este documento se ha presentado una demostración de cómo podría ser un plan de reversión para los controladores de dominio. Es importante destacar que esta demostración es una representación y que, en la práctica, se deben considerar las configuraciones específicas de cada cliente al implementar un plan de marcha atrás.

3.5.4. Nodo físico de un *cluster*

Como se explicó en el punto 3.4.1 un hipervisor puede formar un *cluster* de servidores, agrupando varias máquinas virtuales en dos o más nodos físicos, consiguiendo así una alta disponibilidad y garantizando que aunque pueda quedar inoperativo uno de los nodos que aloja máquinas virtuales, el *cluster* puede proporcionar servicio, asumiendo otro nodo la carga de las máquinas.

Lo primero que hay que hacer es verificar el estado del nodo, si un nodo queda completamente inoperativo, hay que pausar el nodo del *cluster*, quitándole todos los recursos compartidos y roles asociados para evitar problemas y que el *cluster* siga funcionando correctamente. Posteriormente, se debe intentar reparar, ya sea desde una ILO si consta

[29] *Flexible Single Master Operations:* Encargado de hacer cambios en el AD, siendo el DC primario.
[30] Del inglés *Directory Services Restore Mode*.
[31] Del inglés *Domain Name System*.
[32] Del inglés *Update Sequence Number*.

de ella, o en modo seguro. En caso de que no funcionara, se debe expulsar el nodo del *cluster* para aislarlo por completo, este proceso será distinto en función de la tecnología que se esté usando. Tras haber aislado el nodo, se deben revisar bien los errores para ver si hay solución y plantear, intentar recuperar un estado anterior de una posible copia de seguridad o de si es más viable reemplazar el nodo, teniendo la necesidad de adquirir nuevo hardware. Cuando ya se haya restaurado la copia de seguridad o en su lugar reemplazado el nodo, se debe volver a añadir el nodo al *cluster*.

Para que este proceso funcione correctamente, es imprescindible seguir unas buenas prácticas y tener una configuración correcta y precisa del *cluster*, para que en caso de fallo el resto de nodos puedan asumir la carga de forma correcta sin cortes.

CAPÍTULO 4
Ejecución y optimizaciones posibles

Este capítulo trata en profundidad cómo realizar el parcheo de servidores, tras haber realizado la planificación con el cliente y el estudio de las posibles herramientas de automatización.

Se explican los pasos previos al parcheo con sus comprobaciones pertinentes, la manera de proceder mientras se realiza la instalación de los parches y las comprobaciones posteriores que hay que realizar para asegurar el correcto funcionamiento de los servidores y la instalación del parche. Asimismo, se describen diversas opciones para poder optimizar y agilizar los pasos mostrados.

La figura 4.1 muestra un diagrama de flujo que resume la secuencia de decisiones y acciones que deben considerarse para completar con éxito el proceso de parcheo de servidores.

Por último, también se incluye un apartado sobre un caso especial, con el que se requiere seguir paso a paso el proceso descrito para asegurarse su correcto funcionamiento.

4.1 Previo al parcheo

En esta sección, se van a explicar los pasos previos a instalar el parche y las comprobaciones necesarias que se deben hacer. Se va a mostrar como se debe notificar al cliente antes de realizar la intervención, además de varias comprobaciones para asegurar que el parcheo se puede realizar correctamente, como, por ejemplo, revisar que haya espacio de almacenamiento suficiente o que los servicios de Windows estén funcionando correctamente.

4.1.1. Notificación al cliente

Tras haber acordado la planificación con el cliente y tener establecidos los servidores a parchear y ventanas de mantenimiento, se le debe informar cada momento que se vayan a realizar los parcheos de los servidores, tanto antes de empezar para avisar de la indisponibilidad temporal, como al finalizar para realizar las comprobaciones necesarias.

Esto se deberá realizar de la manera acordada con el cliente, lo más común, es mediante correo electrónico. En la figura 4.2 se puede observar un correo típico para el parcheo de servidores de un cliente.

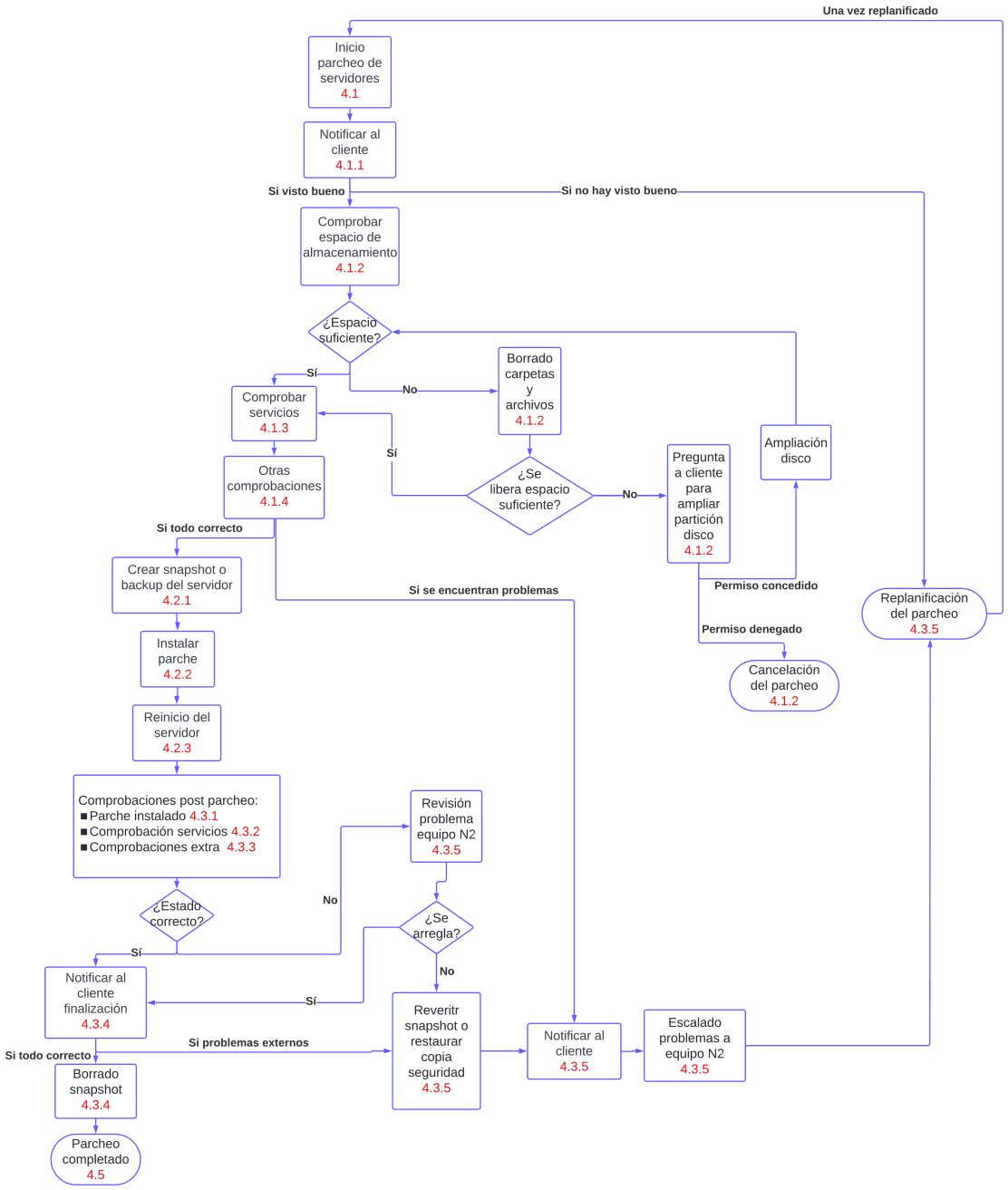
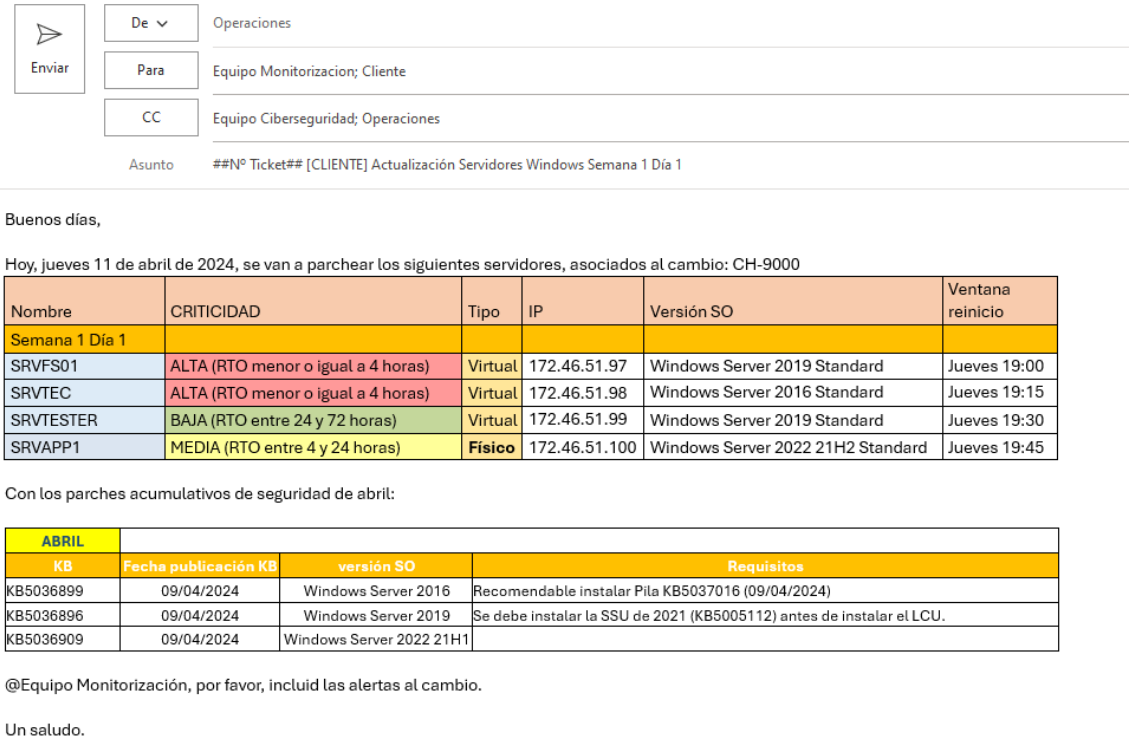


Figura 4.1: Diagrama de flujo para la correcta ejecución del parcheo de servidores. La sección en la que se comenta cada uno de los aspectos considerados se identifica en rojo en el diagrama

El formato mostrado en la imagen 4.2, es un ejemplo de cómo podría ser el correo previo a realizar el parcheo de servidores, aunque los apartados mostrados pueden variar en función de lo que se haya acordado con el cliente. Por ejemplo, en algunos casos quizá requiera una columna extra llamada **Proveedor**, por si en caso de que hubiera algún problema hubiera que contactar directamente con él, o alguna columna de **Requisitos adicionales** si el servidor en cuestión necesitara algún paso adicional ya sea previo o posterior al reinicio. En la tabla donde se indican los servidores de la imagen 4.2 se muestra el nombre, la criticidad (medida en el tiempo máximo tolerable para su recuperación o RTO[33]) que indica el máximo tiempo de inactividad del servidor para no ocasionar problemas o pérdidas en la organización. También se puede observar el tipo de servi-

[33] Del inglés: *Recovery Time Objective*.

dor, en este caso son virtuales y uno físico (no hay ninguno especial, como pudiera ser un nodo de un hipervisor), la versión del SO, la dirección IP[34] y la hora de reinicio que tiene establecida. El correo se envía desde el equipo de operaciones, avisando al cliente y al equipo de monitorización para que estén atentos de posibles alertas o errores que puedan aparecer durante la ejecución.



Enviar | De: Operaciones | Para: Equipo Monitorizacion; Cliente | CC: Equipo Ciberseguridad; Operaciones | Asunto: ##Nº Ticket## [CLIENTE] Actualización Servidores Windows Semana 1 Día 1

Buenos días,

Hoy, jueves 11 de abril de 2024, se van a parchear los siguientes servidores, asociados al cambio: CH-9000

Nombre	CRITICIDAD	Tipo	IP	Versión SO	Ventana reinicio
Semana 1 Día 1					
SRVFS01	ALTA (RTO menor o igual a 4 horas)	Virtual	172.46.51.97	Windows Server 2019 Standard	Jueves 19:00
SRVTEC	ALTA (RTO menor o igual a 4 horas)	Virtual	172.46.51.98	Windows Server 2016 Standard	Jueves 19:15
SRVTESTER	BAJA (RTO entre 24 y 72 horas)	Virtual	172.46.51.99	Windows Server 2019 Standard	Jueves 19:30
SRVAPP1	MEDIA (RTO entre 4 y 24 horas)	Físico	172.46.51.100	Windows Server 2022 21H2 Standard	Jueves 19:45

Con los parches acumulativos de seguridad de abril:

ABRIL			
KB	Fecha publicación KB	versión SO	Requisitos
KB5036899	09/04/2024	Windows Server 2016	Recomendable instalar Pila KB5037016 (09/04/2024)
KB5036896	09/04/2024	Windows Server 2019	Se debe instalar la SSU de 2021 (KB5005112) antes de instalar el LCU.
KB5036909	09/04/2024	Windows Server 2022 21H1	

@Equipo Monitorización, por favor, incluid las alertas al cambio.

Un saludo.

Figura 4.2: Correo para notificar al cliente y equipos implicados previo al parcheo de servidores

Además, también se ve en la imagen 4.2 una tabla con los parches que se van a instalar para los distintos SO con los requisitos que tiene cada parche. También se hace referencia al cambio CH-9300, esto es una alusión a la herramienta de *ticketing* para tener presente y controlado los activos y la hora de la intervención.

Puede darse el caso de que tras enviar este correo, se reciba una respuesta del cliente, indicando que se debe replanificar el parcheo. En este punto no se podrá hacer nada más, ya que la replanificación se deberá a problemas externos o internos de su organización. Únicamente se procederá a fijar otra fecha.

4.1.2. Comprobación de espacio de almacenamiento disponible

Para que la instalación del parche se realice de forma correcta, es necesario que haya cierto espacio de almacenamiento o no se podrá llevar a cabo la instalación, saturando el almacenamiento del servidor y fallando la instalación del parche. Por eso, existen diversas opciones para intentar liberar espacio en el servidor.

- **Carpeta *SoftwareDistribution***: si esta carpeta ocupa demasiado en el servidor, se puede prescindir de ella, ya que la carpeta almacena archivos temporales generados por Windows Update para las actualizaciones pendientes e instaladas. Para poder vaciarla correctamente primero hay que parar el servicio de Windows Update

[34] Del inglés: *Internet Protocol*.

(*wuauserv)* y el servicio de transferencia inteligente en segundo plano (*BITS*). Posteriormente, se puede eliminar el contenido de la carpeta o la carpeta en sí, puesto que luego se generará automáticamente. La ruta es *C:\Windows\SoftwareDistribution*. Después de haber eliminado el contenido y haber liberado el espacio de almacenamiento pertinente, se deben iniciar los dos servicios parados anteriormente. [35]

- **Limpieza de disco**: esta herramienta sirve para eliminar archivos prescindibles que se han almacenado en disco. Para ello se puede ejecutar con el comando *Cleanmgr.exe* o desde el apartado de **Herramientas administrativas de Windows**, clicaremos sobre **Liberador de espacio en disco** seleccionando en este caso la unidad **C:** u otra si se quisiera limpiar también. Como se observa en la imagen 4.3, están marcados archivos que Windows detecta como innecesarios, dando una pequeña descripción de lo que se va a eliminar, pudiendo marcar o desmarcar la casilla en caso de querer eliminarlo o no.

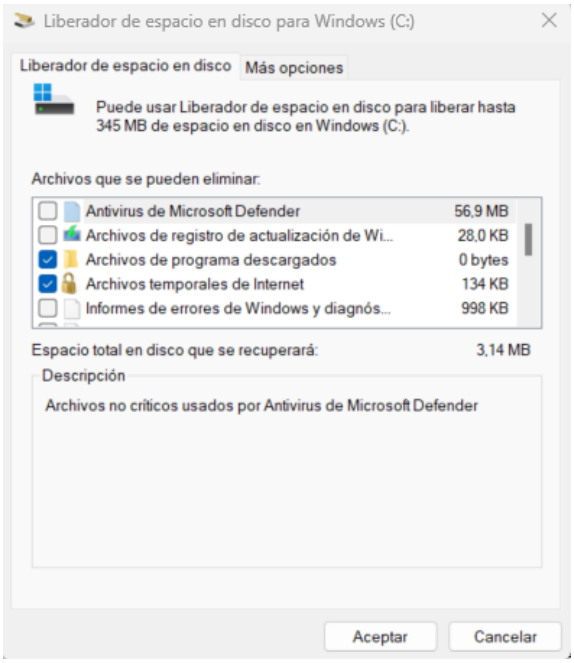


Figura 4.3: Limpieza de la unidad **C**

- **Archivos temporales**: puede darse el caso de que Windows vaya almacenando demasiados archivos temporales, de los cuales algunos pueden estar dañados o corruptos, provocando problemas con algunos programas y llenando el espacio de almacenamiento. Para solucionar esto, desde el panel de ejecutar se escribirá *%tmp%* y se seleccionarán todos los archivos y carpetas para su posterior eliminación.

- **Eliminar carpetas de usuario**: este apartado va a describir el caso de que haya carpetas de usuarios que ya no formen parte de la organización, o sean usuarios obsoletos dado que ya no se accede con esa cuenta, pero no se haya procedido a su borrado y sigan consumiendo espacio de almacenamiento debido a tener documentos, instaladores o cualquier archivo en su carpeta local. Para proceder a eliminar esas carpetas de usuarios de forma correcta, no es suficiente con eliminar la carpeta, ya que posteriormente, si se requiere usar ese usuario, se quedará en un estado inconsistente, creando un perfil temporal.

[35] Información obtenida de: https://help.wnpower.com/hc/es/articles/360045419171.

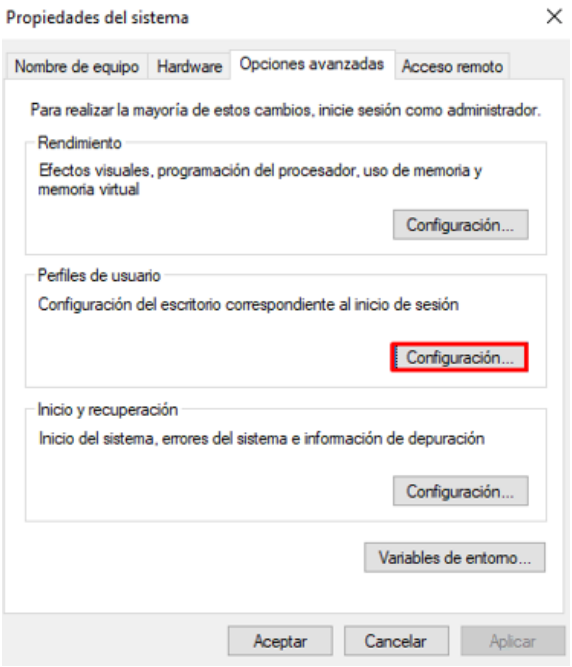


Figura 4.4: Configuración de perfiles de usuario desde la configuración avanzada del sistema

Para liberal espacio de la manera correcta, se debe abrir el panel de control y desde ahí, acceder a **Sistema y Seguridad >Sistema >Configuración avanzada del sistema**. Al pulsar sobre esta última opción, se abrirá una ventana donde aparece la opción de **Perfiles de usuario** tal como se muestra en la figura 4.4. Tras pulsar sobre **Configuración**, se mostrará un listado de todos los usuarios que han iniciado sesión en el servidor, y que tienen una carpeta creada en la ruta *C:\Usuarios*. De todos los perfiles que aparecerán, se seleccionan aquellos que se quieren eliminar, como se aprecia en la imagen 4.5.

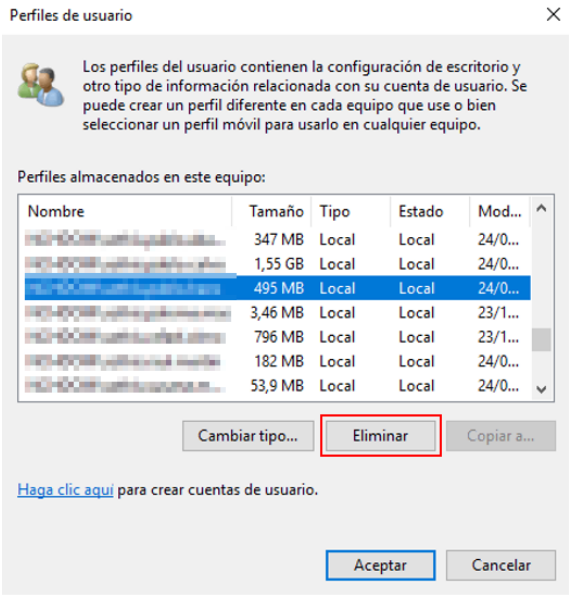


Figura 4.5: Eliminar perfiles de usuario

En caso de que una carpeta de usuario se haya eliminado de forma incorrecta, por ejemplo borrando la carpeta directamente de *C:\Usuarios*, si ese usuario intenta volver a acceder, se encontrará el mensaje de *"No podemos iniciar sesión en la cuenta"*

y se le habrá creado un perfil temporal. Para solventar este error se debe editar el registro de Windows mediante **Regedit**[36]. Desde ahí se buscará la siguiente ruta: **Equipo\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList**. En esa ruta aparecerán carpetas que corresponden a entradas del registro, algunas de ellas con extensión *.bak*, como se ve en la imagen 4.6.

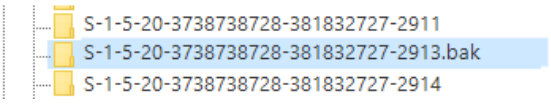


Figura 4.6: Perfiles temporales desde el editor del registro de Windows

Dentro de esas carpetas hay que fijarse en el valor *ProfileImagePath* que es el que indica el nombre de usuario, como se ve en la imagen 4.7.

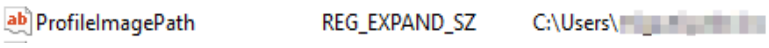


Figura 4.7: Valor *ProfileImagePath* del usuario que se quiere recuperar el perfil

Si el nombre de usuario coincide con el que se desea recuperar el perfil y eliminar su perfil temporal, se debe dar clic derecho sobre la carpeta y eliminarla. Después de este procedimiento, el usuario podrá volver a acceder a su perfil. Para mayor seguridad, en caso de que hubiera algún problema, antes de eliminar la carpeta *.bak* se puede exportar a otra carpeta del equipo a modo de copia de seguridad.

- **Ampliar partición del disco de una máquina virtual**: en caso de que no sea suficiente con eliminar las carpetas o archivos de los pasos anteriores, se le propondrá al cliente realizar la ampliación de la partición del disco. Para poder realizar este apartado, se debe contar con su aprobación, puesto que ampliar el espacio de almacenamiento de una unidad de disco conlleva costes adicionales. Desde el hipervisor o la infraestructura donde estén alojadas las máquinas virtuales, en el apartado de configuración se seleccionará el disco duro virtual que se quiera expandir y se añadirán los *gigabytes* que se soliciten. Para terminar de realizar el proceso, dentro de la propia máquina virtual, se accederá a la **administración de discos**, donde aparecerá en el disco, un espacio adicional no asignado. Se clicará sobre la partición que se quiera ampliar, que esté colindante a ese volumen no asignado, y se seleccionará **Extender volumen**.

En caso de que no hubiera espacio suficiente ni para realizar la instalación del parche por red y el cliente no permita ampliar la partición del disco, el parcheo de este servidor se cancelará. No se podrá replanificar este servidor hasta que el cliente decida aumentar el espacio de almacenamiento disponible.

4.1.3. Comprobación previa de servicios

Antes de reiniciar el servidor, para que se apliquen las actualizaciones correspondientes, se deben comprobar los servicios de Windows para ver aquellos que están iniciados y los que tienen el arranque automático para iniciarse solos después del reinicio del servidor. Como se ve en la figura 4.8 desde **Windows Powershell** ejecutamos el comando

[36]Editor del Registro de Windows. Es la base de datos donde se guardan los ajustes de configuración.

get-service >servicios.txt Para que saque el listado de todos los servicios del servidor y posteriormente poder realizar, junto al **Administrador del servidor**, la comprobación.

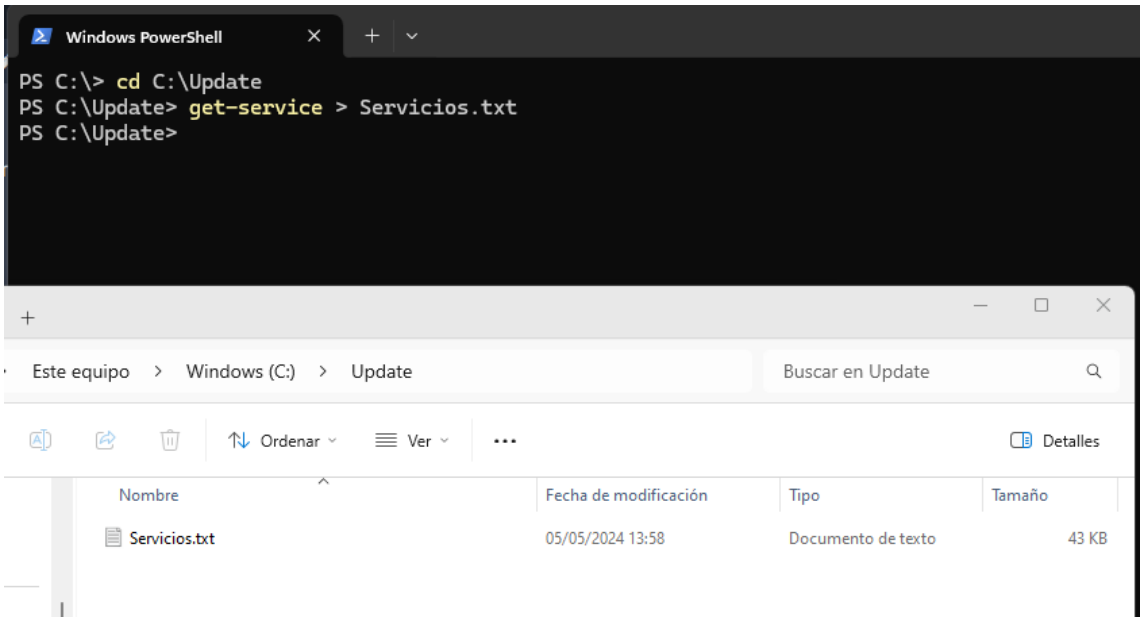


Figura 4.8: Comprobación de servicios de Windows desde **Windows Powershell**

4.1.4. Pasos extra

Este apartado se refiere a comprobaciones extra que se deban hacer antes de realizar el parcheo de un servidor en cuestión. Este paso se debe haber acordado previamente con el cliente o equipos afectados. Puede ser desde, por ejemplo, tener que apagar otros servidores que dependan del que se va a parchear, parar alguna aplicación, pausar alguna copia de seguridad o parar alguna instancia de un servidor de base de datos en caso de que hubiera dependencias.

4.2 Durante el parcheo

En esta sección, se van a explicar los pasos a realizar durante el parcheo de servidores, desde crear una *snapshot* o un *backup*, hasta instalar el parche y reiniciar el servidor. En este punto, ya se han realizado todas las comprobaciones previas, por lo que se puede proceder a parchear los servidores correctamente.

4.2.1. Creación de *snapshot* o *backup*

El primer paso a realizar durante el parcheo, será realizar parte del plan de marcha atrás explicado en la sección 3.5. Para ello, dependiendo del tipo de servidor realizaremos una *snapshot* o un *backup*.

En caso de parchear un servidor físico, se requerirá realizar un *backup* previo al parcheo para tener preparado el plan de marcha atrás. En caso de gestionar el *backup* desde el propio servidor, como muestra la imagen 3.10, sería suficiente con clicar sobre *Backup Now* en caso de utilizar la herramienta **Veeam**. Si el *backup* se realiza desde el servidor de *backup*, se realizará de la forma explicada en la imagen 3.11.

Como se ve en la imagen 3.10, clicando sobre *Backup Now* se crearía un *backup* en ese instante, si se gestiona desde el propio servidor o realizarlo desde el servidor de *backup* como se explicó en la imagen 3.11.

En caso de parchear un servidor virtual, se procederá a crear una *snapshot* desde la herramienta que se esté utilizando. Es importante que a la hora de crear la *snapshot* se marque la casilla de incluir en la *snapshot* la memoria de la máquina virtual, como muestra la imagen 4.9. Ya que, de este modo, se conseguirá retener el estado activo del servidor, manteniendo los procesos y elementos que había abiertos en ese momento. Si no se incluye la memoria de la máquina y se necesita revertir la *snapshot*, cuando se revierta la máquina se apagará y los procesos que estaban abiertos o en ejecución no habrán sido guardados.

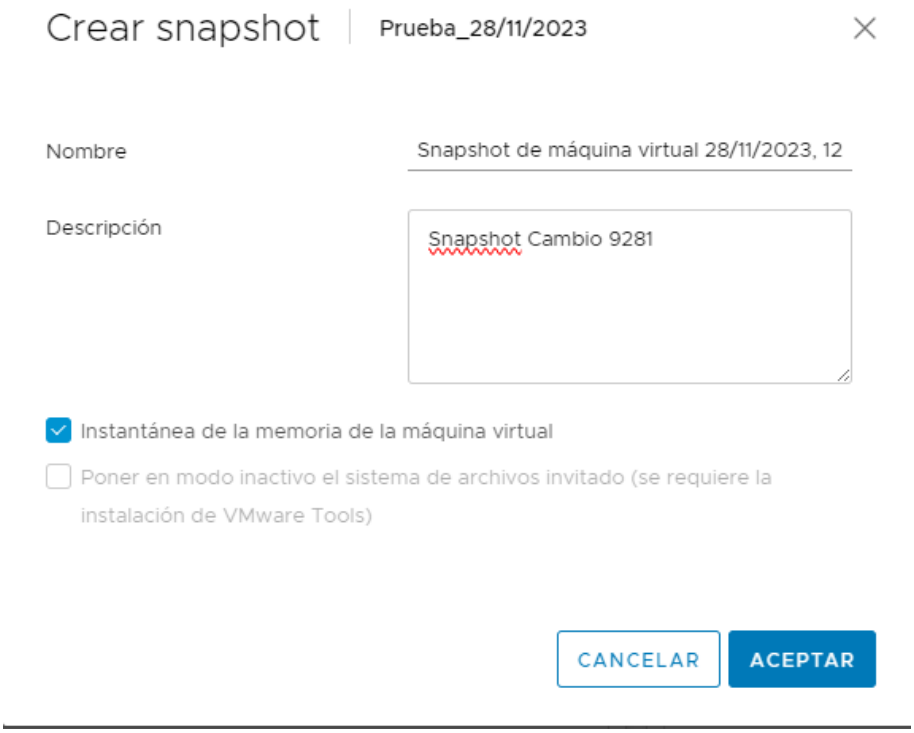


Figura 4.9: *Snapshot* incluyendo la memoria de la máquina virtual

4.2.2. Instalación del parche

Para la instalación del parche se puede hacer de varias formas:

- **Manual**: la forma manual de hacerlo es descargar en local el instalador de Windows Update que tiene una extensión de archivo *.msu* desde el catálogo de **Microsoft**. Este archivo se dejará en la carpeta que se desee, aunque es recomendable crear una carpeta en **C:** llamada *Updates* para tener a mano el instalador y que nadie lo borre por accidente si se deja en descargas o en el escritorio. Posteriormente, le daremos clic derecho y saltará el aviso del instalador independiente de Windows Update. En el ejemplo que muestra la imagen 4.10, se solicita si se desea instalar el parche con identificador **KB5031419** referente al parche de octubre de 2023 de Windows Server 2012R2.

  Cuando se indique que *sí* se desea continuar con la instalación, aparecerá en pantalla el instalador mostrando una barra de progreso de la instalación similar a la de la imagen 4.11. Posteriormente, cuando acabe la instalación, aparecerá la opción de

reiniciar directamente desde esa interfaz o cerrar esa ventana si se desean realizar más operaciones en el servidor.

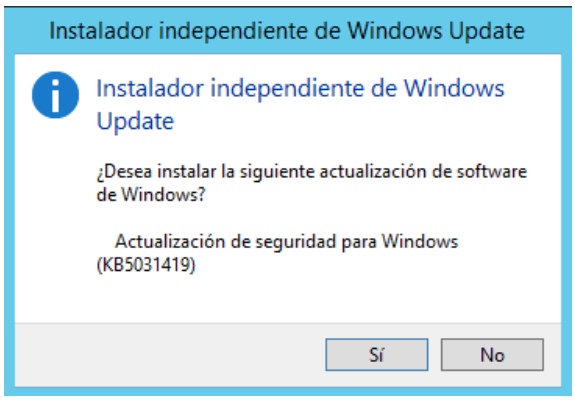


Figura 4.10: Instalador independiente de Windows Update

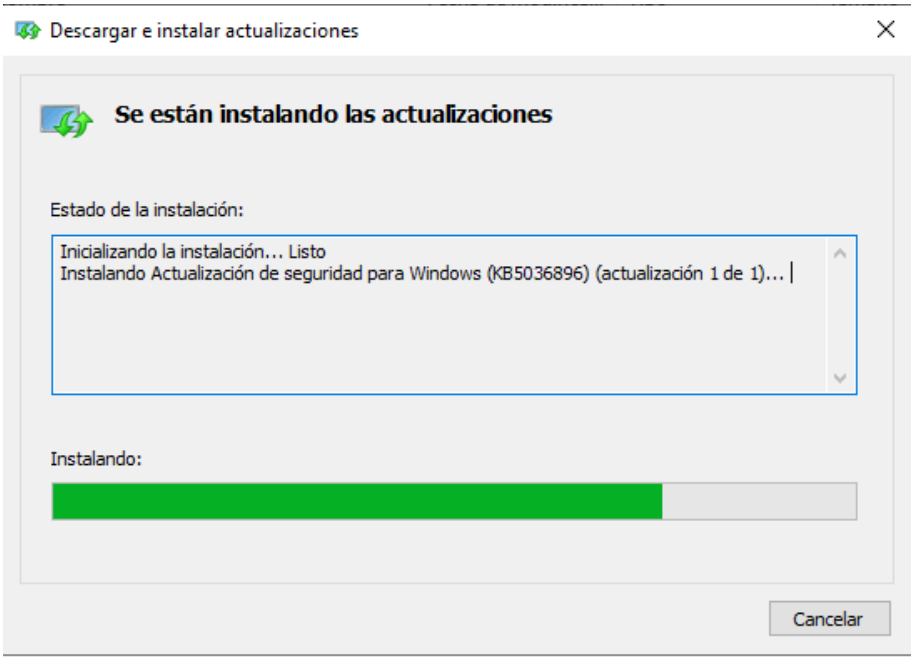


Figura 4.11: Instalación manual de un parche de Windows Update

- **Sconfig**: es una herramienta para la configuración y administración de una única instancia de Windows Server. Se pueden configurar varios aspectos del sistema operativo, no solo en cuanto a la descarga e instalación de actualizaciones de **Microsoft**, sino también para configuraciones del AD, configuraciones de red o más opciones como se ve en la imagen 4.12.

  Respecto al tema de las actualizaciones de Windows Update tenemos dos apartados del **Sconfig** que son el cinco y seis de la imagen 4.12. Para poder usar ambos correctamente y ejecutar todos los comandos se debe usar la herramienta **sconfig** desde una consola **Windows Powershell** o símbolo del sistema de Windows (cmd) con permisos de administrador.

  El quinto apartado sirve para establecer la configuración de las actualizaciones, ya sea que estén establecidas de forma automática para su descarga e instalación, cuando el propio servidor lo decida en función a sus horas de uso habitual, o de forma manual, para que sea el administrador el que realice la instalación cuando sea

más conveniente. Este cambio se aprecia en la imagen 4.12 como, al escribir el número *5*, y posteriormente la letra *m*, se deshabilitan las actualizaciones automáticas y se activan de forma manual.

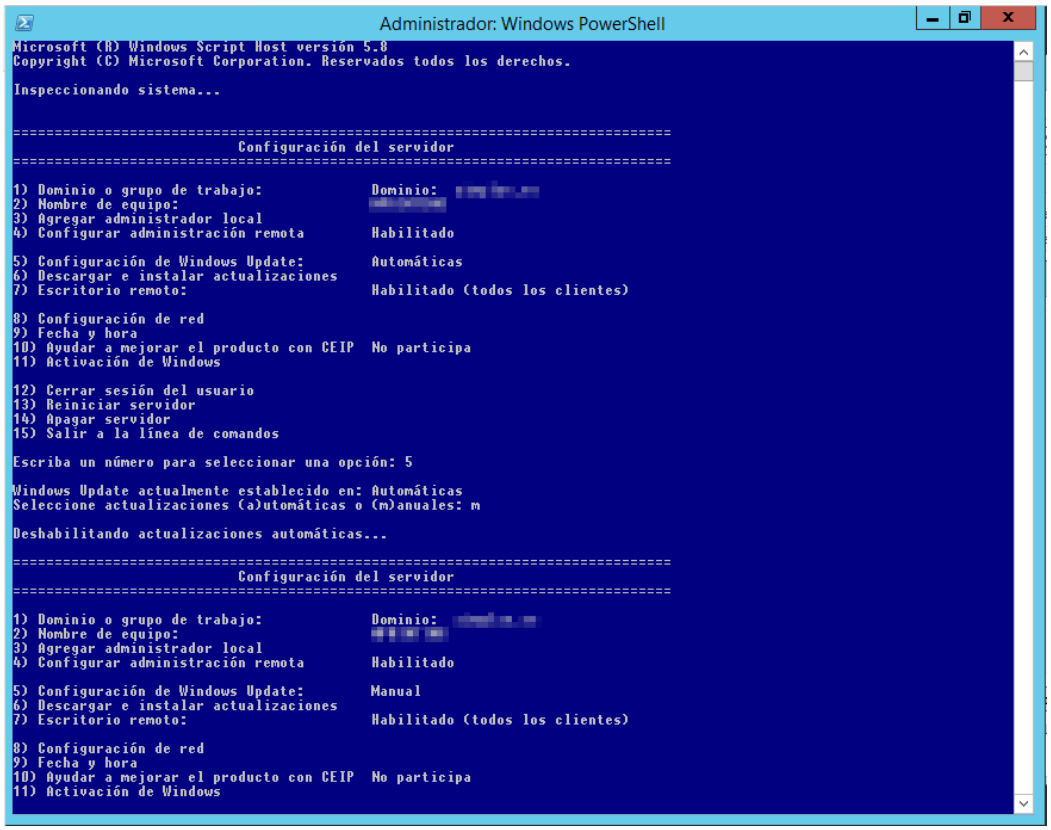


Figura 4.12: Opciones mostradas para configurar desde **Sconfig**

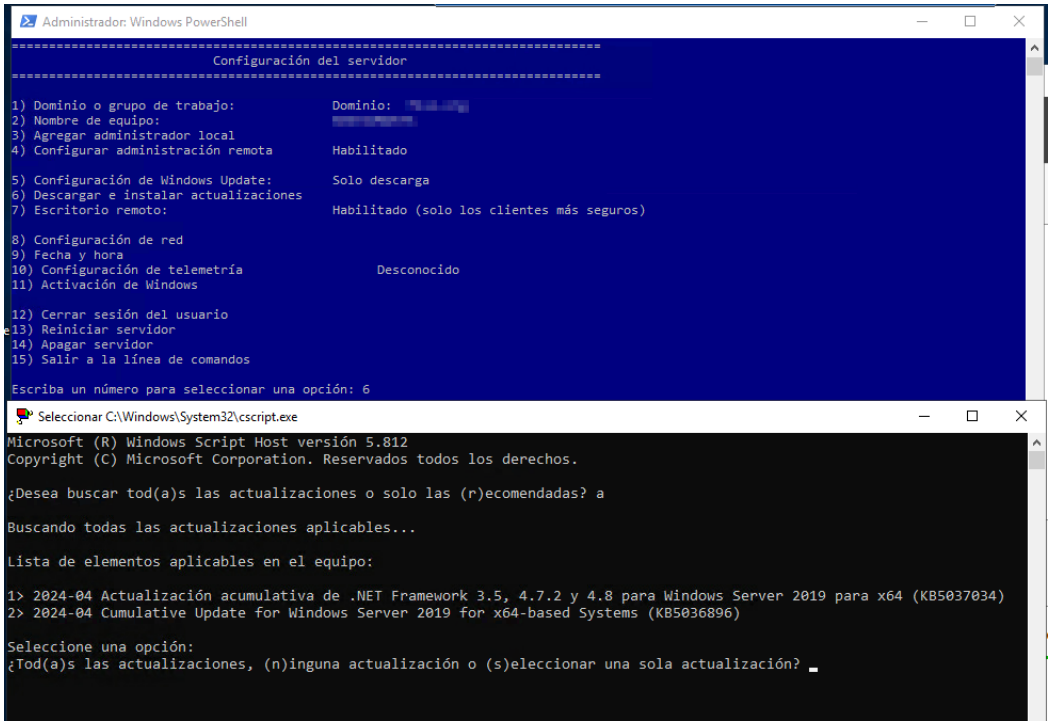


Figura 4.13: Actualización de Windows desde el menú de **Sconfig**

El sexto apartado sirve para poder descargar e instalar las actualizaciones por red. Esta característica es de gran utilidad cuando se requiere realizar el parcheo manual, pero queda poco espacio en el servidor, por lo que de esta forma se ahorra el tener que copiar el parche de aproximadamente un 1GB[37] de tamaño. Como se ve en la imagen 4.13 al marcar el número *6* se abre una nueva ventana donde pregunta que actualizaciones se desean: buscar entre todas las posibles (*a*) o solo las recomendadas (*r*). Al seleccionar una de las dos opciones, se generará un listado con las actualizaciones recomendadas o todas en función de lo que se haya seleccionado. En este punto se pueden instalar todas (*t*), ninguna (*n*) o seleccionar una única (*s*).

Si se quisiera profundizar más las opciones que ofrece *Sconfig*, se debe consultar la página web de **Microsoft Learn** [26].

- **Automatizado**: ese punto se refiere a realizar la instalación del parche usando una herramienta de automatización como las explicadas en la sección 5.4.

4.2.3. Reinicio del servidor

Después de la instalación del parche, se necesitará un reinicio del servidor para que se termine de aplicar la actualización. Este reinicio se puede hacer de diversas formas.

- **Manual**: en caso de que se requiera de una persona activamente para reiniciar el servidor y realizar las comprobaciones pertinentes justo nada más se reinicie, el servidor se reiniciará de forma manual desde el propio servidor pulsando sobre el icono de Windows y pulsando en reiniciar.
- **Tarea programada**: en caso de que el parcheo se realice fuera de horas y no sea crítico estar justo después para realizar las comprobaciones o simplemente para no tener que estar pendiente justo a la hora exacta para reiniciarlo, se puede crear una tarea programada de reinicio desde el **Programador de tareas de Windows**. En caso de que un servidor se quiera reiniciar cada cierto tiempo a la misma hora, se puede crear una tarea programada que se repita diaria, semanal, mensualmente o según la frecuencia deseada. La imagen 4.14 muestra una tarea programada para que cada dos miércoles a las seis de la mañana se reinicie el servidor.

  Se pueden ejecutar diversas acciones desde el programador de tareas. Una de ellas es reiniciar el servidor, como se observa en la figura 4.15, usando el ejecutable *shutdown.exe* con los parámetros */g /f /t 60*.

  - */g* cierra sesión y reinicia el servidor, tras encenderse, reiniciar las aplicaciones.
  - */t 60* significa el tiempo de espera en segundos usado para advertir a los usuarios que estén dentro del equipo para que guarden su trabajo antes de que se reinicie.
  - */f* fuerza el cierre de las aplicaciones, esto puede ser esencial, ya que a veces una aplicación que no se cierra correctamente puede provocar que el servidor no se reinicie debido a esa aplicación que se ha quedado bloqueada.

  Existen muchas más opciones que se pueden configurar. Estas se pueden listar ejecutando el comando *shutdown* desde la consola. Además de reiniciar el servidor, se puede añadir a la lista de acciones un archivo *.bat* o archivo por lotes, para que se ejecuten una serie de comandos antes de que se reinicie el servidor como, por ejemplo, pausar algún servicio que requiera ser parado antes del reinicio.

[37] GygaByte.

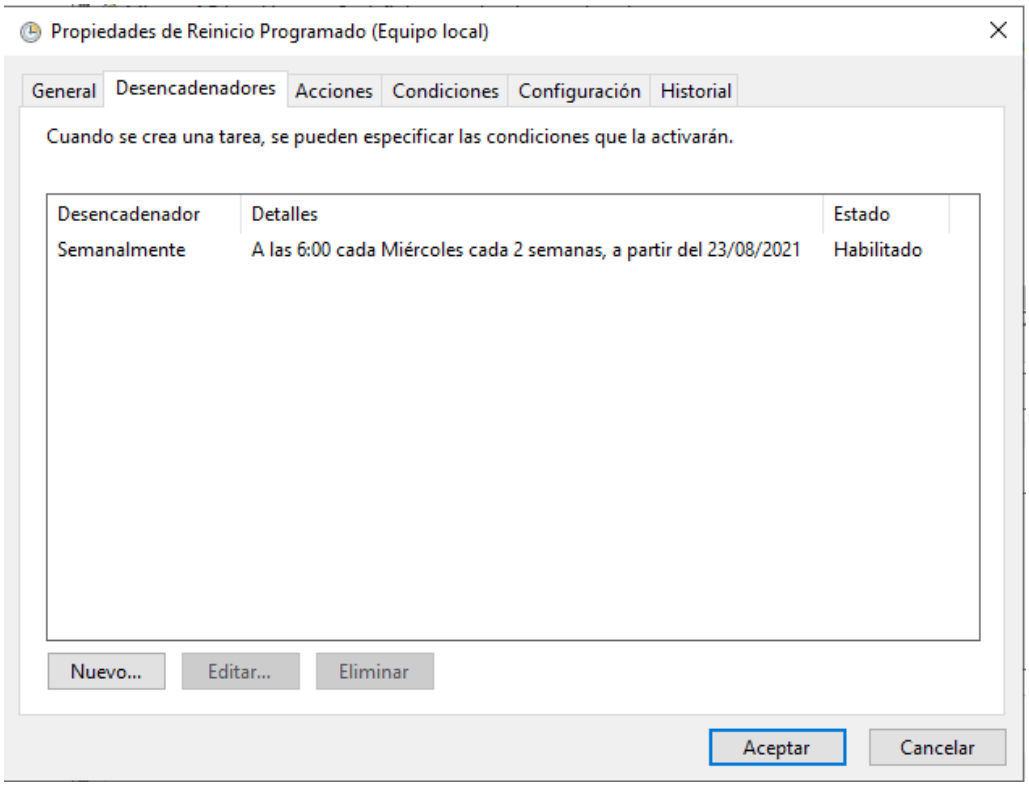


Figura 4.14: Configuración de la frecuencia de ejecución de una tarea programada

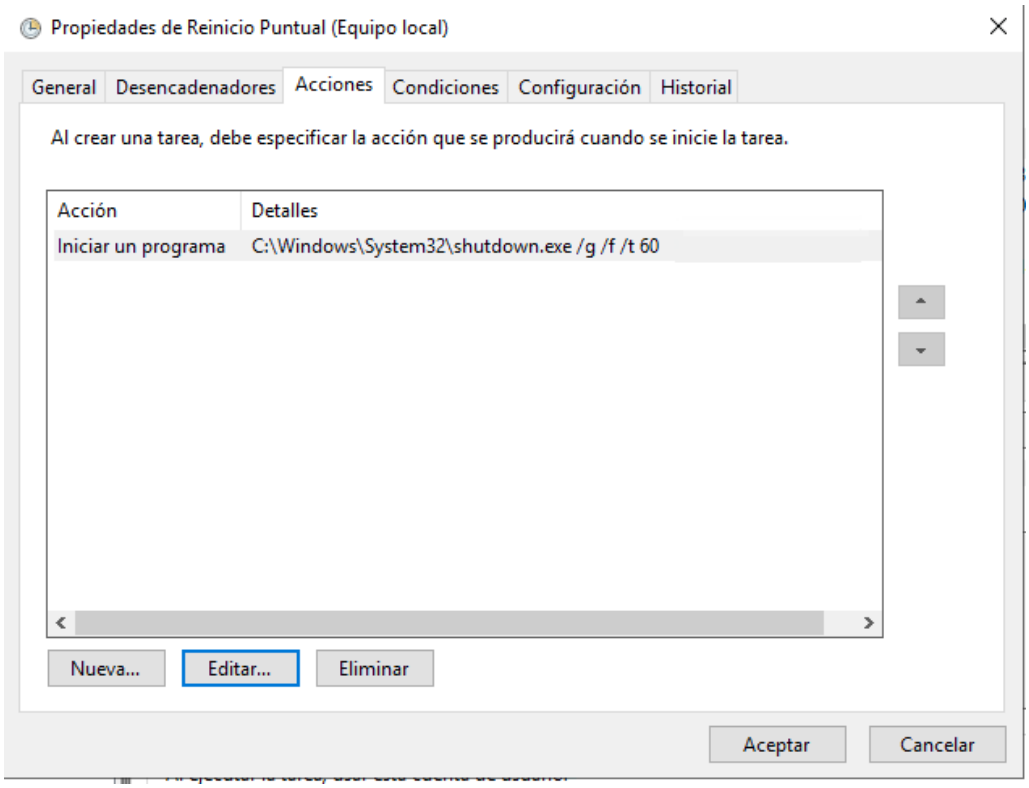


Figura 4.15: Parámetros y acciones para reiniciar un servidor mediante una tarea programada

- **Automatizado**: este punto se refiere a reiniciar el servidor, usando la herramienta de automatización que se haya utilizado para la instalación de los parches.

4.3 Posterior al parcheo

Esta sección hace referencia a los pasos a seguir tras haber instalado el parche y reiniciado el servidor.

4.3.1. Comprobación posterior de servicios

Después del reinicio, se debe verificar que los servicios que se obtuvieron previamente en el apartado 4.1.3, estén igual. Esto quiere decir que los servicios que debían estar apagados permanezcan así y que los que tengan un inicio automático se inicien correctamente. Habría que comprobar si algún servicio que estuviera activo, tuviera un inicio manual y se tuviera que activar manualmente. Para este apartado se podrá utilizar como herramienta de ayuda el Administrador del servidor, ya que aparecerá un recuadro con eventos, servicios o el rendimiento del servidor. En caso de que haya algo incorrecto estará marcado en rojo con un número indicando los problemas encontrados, como se ve en la imagen 4.16. En este caso se observa que hay tres servicios que están detenidos que son automáticos. Para comprobar si deben estar parados o se debe a algún problema causado por el reinicio o el parche se tendrá que revisar el archivo *servicios.txt* que se obtuvo en la sección 4.1.3 y realizar la comprobación mediante el nombre del servicio. En este caso, el servicio *MySQL* ya estaba parado anteriormente, por lo que el estado es correcto. En caso de que no lo fuera, desde el propio administrador de tareas, pulsando con el botón derecho encima del servicio deseado se podrá iniciarlo.

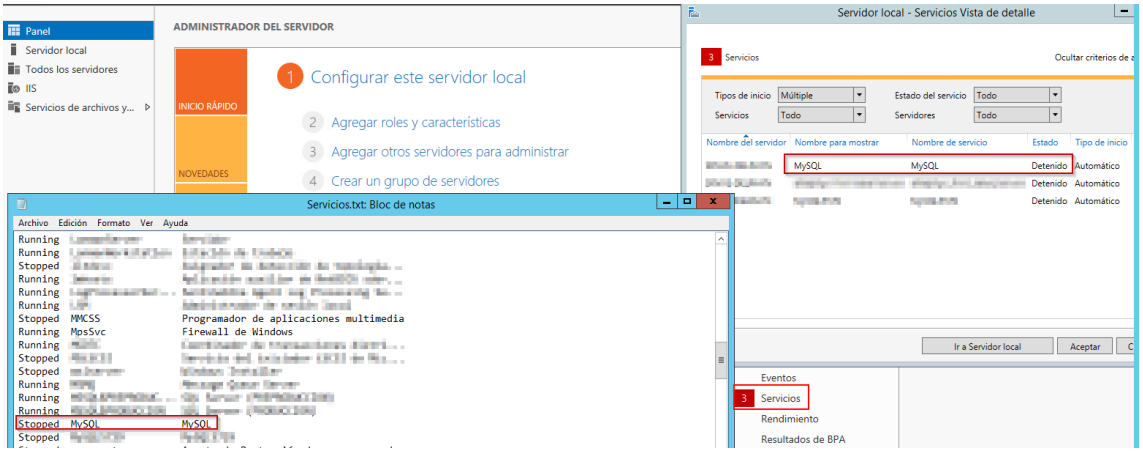


Figura 4.16: Comprobación de servicios mediante el administrador de tareas

4.3.2. Comprobación parche instalado

Este apartado se puede hacer de diversas maneras, aunque siempre es conveniente revisarlo de dos formas, para asegurarse completamente de que la actualización está correctamente instalada:

- **Windows Powershell**: para comprobar desde **Windows Powershell** que el parche se ha instalado correctamente, se podrá escribir uno de los dos siguientes comandos mostrados en la imagen 4.17. Con el comando *Get-HotFix | sort InstalledOn -Descending* se logra ordenar las actualizaciones más recientes instaladas en el equipo ordenadas de forma descendente basándose en su fecha de instalación, pudiendo verificar instalaciones en más de un servidor o filtrar por su descripción. Usando *findstr* se busca directamente si el parche en cuestión se encuentra instalado en el

equipo. En la imagen 5.17, por ejemplo, se busca si el parche KB5036893 se encuentra instalado.

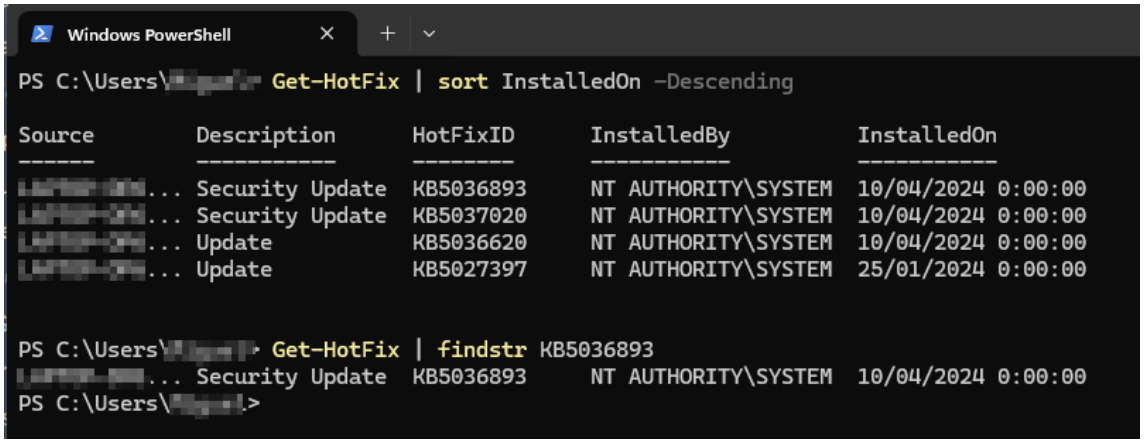


Figura 4.17: Comprobación de la instalación del parche mediante **Windows Powershell**

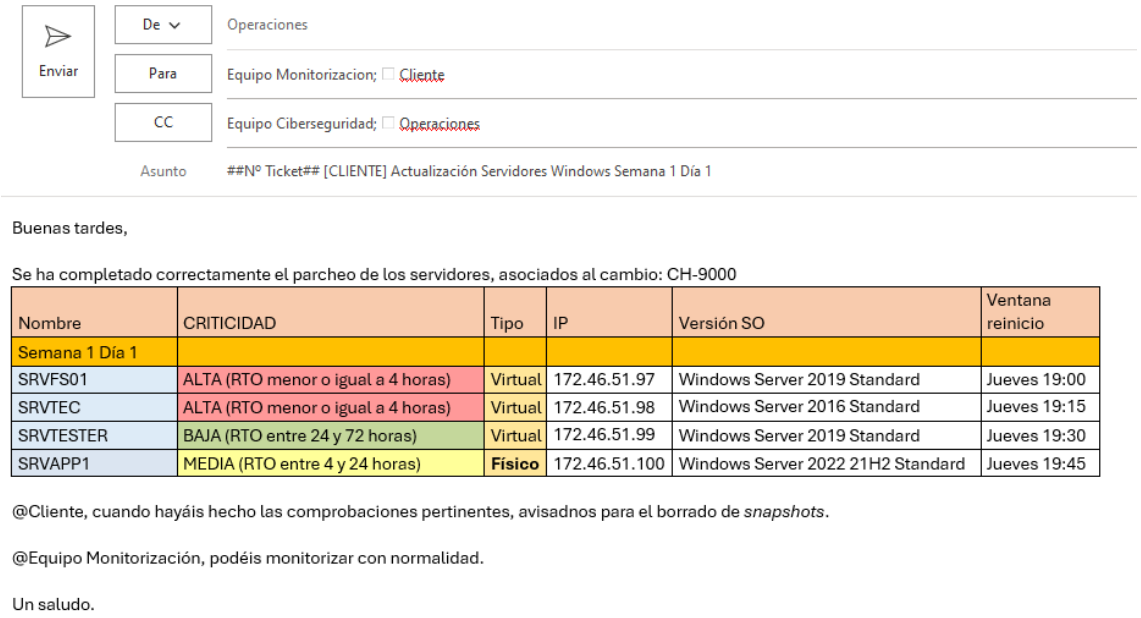
- **Panel de control**: desde el panel de control se accederá al apartado de **Programas** y, posteriormente, en **Programas y características**, se encontrará la opción de **Ver actualizaciones instaladas**. Desde ahí también se podrá comprobar si los parches están instalados correctamente.
- **Configuración de Windows Update**: otro método es directamente desde la configuración de Windows Update, revisar el **Historial de actualizaciones** donde se muestra la fecha de instalación, el parche instalado y un comentario donde indica si está correctamente instalado o no.

4.3.3. Otras comprobaciones

Otras comprobaciones que se deben realizar en el servidor para verificar su correcto funcionamiento incluyen revisar que el consumo de los recursos de la máquina (consumos de memoria, cpu, red...) es estable, que el visor de eventos de Windows no muestre ningún error crítico o discrepancia y, en función del tipo de servidor, se puede requerir revisar algunas aplicaciones o comprobación de que los *backups* funcionan correctamente.

4.3.4. Notificación al cliente

Tras haber comprobado que los servicios funcionan correctamente y los parches están instalados, se debe avisar al cliente para que haga comprobaciones por su parte o por parte de sus proveedores, de que las aplicaciones o dependencias asociadas a esos servidores funcionan correctamente, y que posteriormente nos lo notifiquen para poder borrar las *snapshots* o, si hubiera fallado algo, revertirlas. También se notifica al equipo de monitorización, como se ve en la imagen 4.18, para que esté al tanto de que se ha terminado la intervención y pueda monitorizar todos los activos con normalidad, por si hay cualquier alerta tratarla de inmediato.



Enviar | De: Operaciones | Para: Equipo Monitorizacion; Cliente | CC: Equipo Ciberseguridad; Operaciones | Asunto: ##Nº Ticket## [CLIENTE] Actualización Servidores Windows Semana 1 Día 1

Buenas tardes,

Se ha completado correctamente el parcheo de los servidores, asociados al cambio: CH-9000

Nombre	CRITICIDAD	Tipo	IP	Versión SO	Ventana reinicio
Semana 1 Día 1					
SRVFS01	ALTA (RTO menor o igual a 4 horas)	Virtual	172.46.51.97	Windows Server 2019 Standard	Jueves 19:00
SRVTEC	ALTA (RTO menor o igual a 4 horas)	Virtual	172.46.51.98	Windows Server 2016 Standard	Jueves 19:15
SRVTESTER	BAJA (RTO entre 24 y 72 horas)	Virtual	172.46.51.99	Windows Server 2019 Standard	Jueves 19:30
SRVAPP1	MEDIA (RTO entre 4 y 24 horas)	**Físico**	172.46.51.100	Windows Server 2022 21H2 Standard	Jueves 19:45

@Cliente, cuando hayáis hecho las comprobaciones pertinentes, avisadnos para el borrado de *snapshots*.

@Equipo Monitorización, podéis monitorizar con normalidad.

Un saludo.

Figura 4.18: Correo para notificar al cliente y equipos implicados posterior al parcheo

4.3.5. Resolución de problemas y replanificación del parcheo

En caso de que haya algún problema tras la instalación del parche, se debe escalar el problema al equipo N2. Este equipo se mencionó en la sección 3.1.2 y serán los encargados de intentar resolver el problema de una forma rápida para evitar la indisponibilidad del servidor. En caso de no ser posible su resolución, se debe revertir la *snapshot* o restaurar el *backup* y documentar el problema por parte del equipo N2 para evitar que vuelva a suceder el error, o en caso de que vuelva a ocurrir, poder actuar rápidamente.

En caso de que se haya tenido que revertir la *snapshot*, restaurar el *backup*, o no se pudiera parchear el servidor por problemas previos, se deberá avisar al cliente de lo ocurrido y fijar una fecha para replanificar el parcheo, tras haber solucionado el problema, o haber fijado un método de acción para cuando se vuelva a parchear.

4.4 Casos especiales

- **Parcheo nodo Hyper-V**: el primer paso consiste en comprobar que hay suficientes recursos en los nodos a los que se pretende migrar todas las máquinas virtuales del nodo que se va a parchear. En caso afirmativo, el siguiente paso es realizar la migración de las máquinas virtuales a esos otros nodos desde el *Failover Cluster Manager*[38]. Antes de empezar con este proceso, es aconsejable obtener capturas de pantalla de cómo estaban organizados los nodos antes de realizar la migración, para volver a colocar las máquinas en sus respectivos nodos al finalizar el parcheo.

  Dentro del *Failover Cluster*, en el apartado de roles estarán listadas todas las máquinas y se podrán ordenar por *Owner Node* para identificar en que nodo está cada servidor. Una vez listados y ordenados, se clicará con el botón derecho sobre el servidor que se quiera mover, y se seleccionará la opción *Move*, *Live Migration*. En caso de tener más de dos nodos y querer que el propio **Hyper-V** balancee la carga, se seleccionará al nodo que considere la opción de *Best Possible Node* o en caso de

[38] Gestiona los nodos y proporciona la alta disponibilidad en **Hyper-V**.

querer elegirlo de forma manual se seleccionará *Select Node*. A modo de ejemplo, en la figura 4.19 se observa como una máquina que está ejecutándose sobre el nodo **SRV-HYP02** se selecciona para su migración al nodo **SRV-HYP01**.

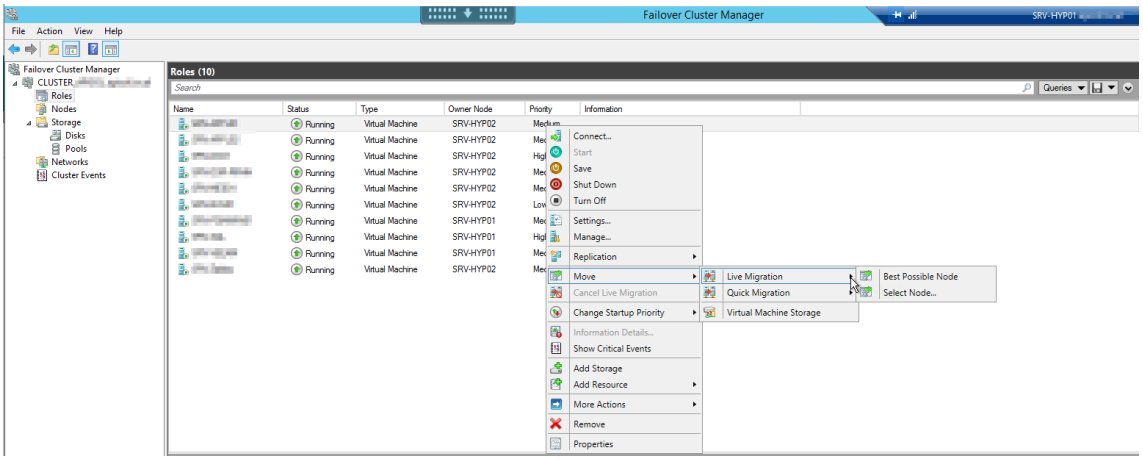


Figura 4.19: *Live Migration* de una máquina virtual desde el *Failover Cluster Manager*

Si hay máquinas apagadas en el nodo, no pueden migrarse con *Live Migration*. Entonces hay dos posibilidades:

1. Utilizar *Quick Migration* que, aunque normalmente produce un pequeño corte en la MV seleccionada, no producirá ese efecto al estar apagada.
2. Dejar la máquina en el nodo, pero asegurándose de que no tenga el encendido automático activo (ver figura 4.20).

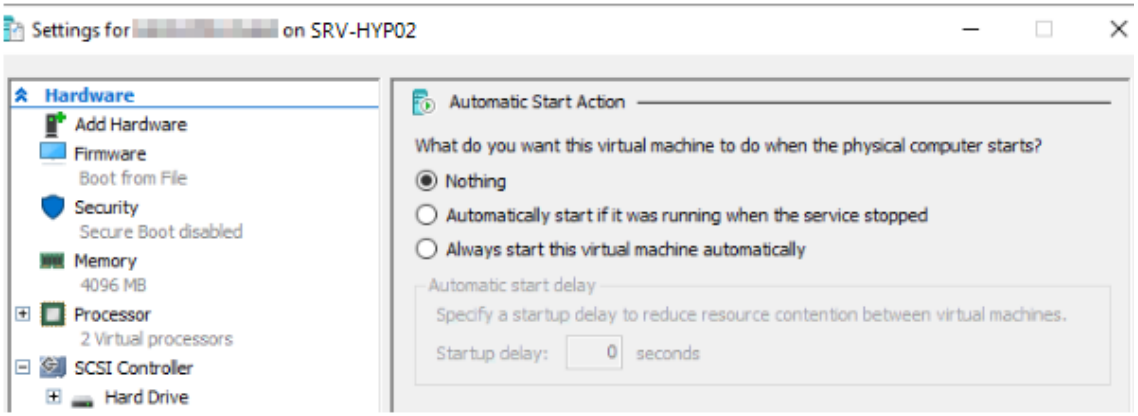


Figura 4.20: Configuración de una máquina virtual para desactivar el inicio automático

En caso de que haya alguna máquina que pertenezca al nodo que se va a parchear, pero no esté agrupada en el *cluster*, se puede migrar a otro nodo del **Hyper-V** desde el *Hyper-V Administrator*. Se seleccionará la opción de mover la máquina virtual a otro **Hyper-V** que esté encendido, como se observa en la figura 4.21. Seleccionando esa opción en lugar de la de mover el almacenamiento de la máquina virtual. Después de elegir la máquina que se va a querer migrar y elegir su destino, se selecciona la opción de mover únicamente la máquina virtual.

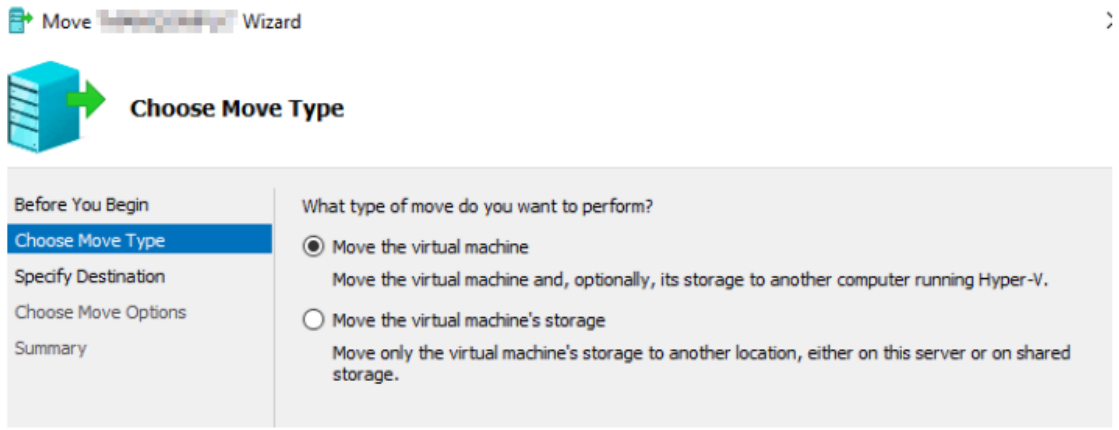


Figura 4.21: Mover una máquina que no está en el *cluster* a otro **Hyper-V**

Tras haber migrado todas las máquinas virtuales a otros nodos, se procederá a migrar los discos que estén actualmente asociados al nodo que se va a parchear. Los discos *Cluster Shared Volume*[39] se moverán automáticamente si los están usando únicamente una de las máquinas migradas, sin embargo, si lo están usando diferentes máquinas, habrá que moverlos de forma manual, siguiendo el mismo procedimiento que el explicado para mover las máquinas a otro nodo.

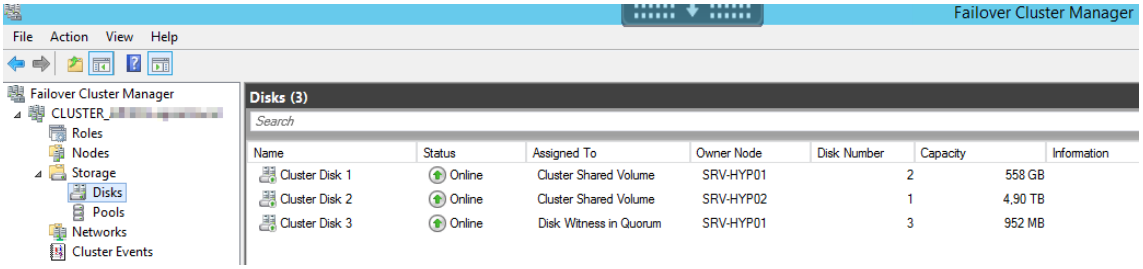


Figura 4.22: Mover discos asociados a los nodos del **Hyper-V**

El disco de *disk Witness in Quorum*, como se ve en la imagen 4.22, es el disco que se denomina disco de testigo. En otras palabras, el nodo que posee ese disco es el que tiene actualmente el rol principal o activo del cluster, mientras que el resto están en un plano secundario. Para poder parchear la máquina correctamente, primero se debe poner en pausa el nodo correspondiente y purgarle los roles que tenga que se pasarán automáticamente a otro nodo. Esto se hará desde el panel de *Nodes* mostrado en la figura 4.23, donde se seleccionará el nodo a parchear y dándole con el botón derecho del ratón, a *Pause* y en el desplegable que aparecerá marcar la opción de *Drain Roles* como se muestra en la imagen 4.24.

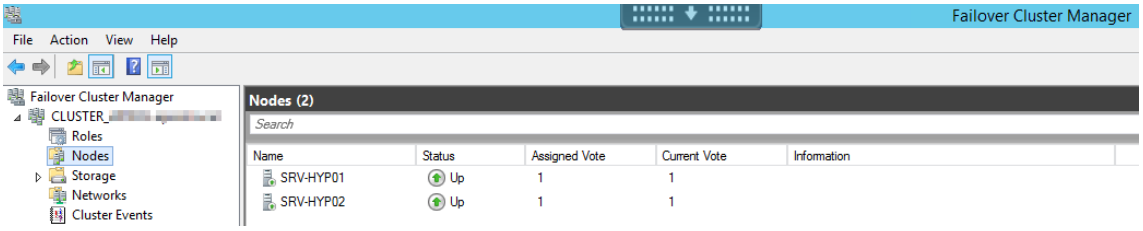


Figura 4.23: Panel de nodos **Hyper-V**

[39] Discos accesibles para lectura y/o escritura de todos los nodos.

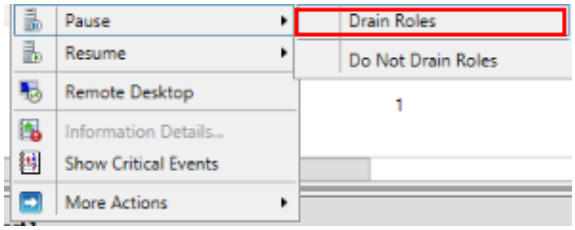


Figura 4.24: Pausar el nodo correspondiente y sus roles asociados

Posteriormente, se seguirán los pasos normales del parcheo de servidores explicados en este capítulo y tras acabar y comprobar el correcto funcionamiento del nodo, en el apartado de *Nodes*, se clicará sobre el nodo pausado, dándole a *Resume* y devolviéndole los roles que tuviera anteriormente con *Drain Roles Back*.

4.5 Conclusiones

En este capítulo se han explicado todas las acciones necesarias para realizar el parcheo de servidores correctamente. Se han explicado los pasos previos al parcheo, donde se explica cómo se notifica al cliente y ciertas comprobaciones que se deben hacer antes de parchear. También, se han detallado las acciones a realizar durante el parcheo de servidores, con distintas formas de optimizar el parcheo y, por último, se han explicado los pasos posteriores que se deben seguir para comprobar que el parcheo ha sido exitoso, o en su defecto cómo actuar si hubiera problemas.

Por último, he de destacar la importancia de disponer de una guía que permite coordinar los diferentes pasos en el parcheo, siendo de gran utilidad para agilizar procesos y no olvidar ningún punto clave durante el parcheo de servidores. También mencionar la utilidad del diagrama 4.1 para tener una primera impresión visual sobre el parcheo y cómo es el flujo de información y acciones que deben suceder. Con estas herramientas, una persona inexperta en el tema será capaz de entender el parcheo de servidores, y para un gestor de informática, le será de gran utilidad para potenciar sus conocimientos y aplicar la metodología.

CAPÍTULO 5
Herramientas y automatizaciones posibles

En este capítulo se van a detallar distinas herramientas que van a ser de gran utilidad a la hora del parcheo de servidores. El uso de estas herramientas va a permitir un mayor control y una mejor estructuración dentro de la organización, consiguiendo automatizar varios procesos. El capítulo se va a dividir en cuatro secciones, donde en cada sección se va a detallar un tipo determinado de herramientas.

En primer lugar, se van a explicar herramientas de *ticketing* para facilitar la organización dentro de una organización y facilitar la gestión de incidencias o rutinas.

En segundo lugar, se van a explicar herramientas para administrar conexiones remotas y de esta forma poder acceder a la infraestructura del cliente y a sus servidores.

En tercer lugar, se van a explicar herramientas de monitorización, para tener controlados los activos en todo momento y poder actuar en cuanto hubiera un problema.

Por último, se van a explicar herramientas para la administración y gestión de parches, consiguiendo así automatizar el proceso del parcheo de servidores.

5.1 Herramientas de *ticketing*

Una herramienta de *ticketing* es, una herramienta de gestión de servicios de TI, fundamental para cualquier organización que necesite una gestión optimizada de las interacciones con los clientes, como ya se comentó en la sección 3.1. Un *ticket* es un elemento dentro de la herramienta de *ticketing* el cual se detalla una petición de servicio, una rutina o una incidencia, además de incluir otra información como la criticidad, o quién lo solicita, entre otras cosas.

De esta forma, se logra estructurar toda la información de los activos del cliente y contratos. También se pueden crear *tickets* para que las rutinas de los parcheos de los servidores, se generen de forma automática según la periodicidad establecida en el contrato. Además, se pueden crear cambios para detallar el proceso de cada parcheo y que todos los equipos implicados estén al tanto. Por lo tanto, estas herramientas, agilizan y facilitan el trabajo a las organizaciones. En la tabla 5.1 se muestran varias herramientas de *ticketing* actuales.

Tabla 5.1: Comparación herramientas de *ticketing*

Herramienta	Características	Licencia
Freshdesk[40]	Proporciona una gestión eficiente de los *tickets*. Permite manejar *tickets* de múltiples canales desde un solo lugar y ofrece gestión de SLA, entre otras funcionalidades.	Comercial
OsTicket[41]	Centraliza las consultas de los clientes en una plataforma web fácil de usar.	Código abierto
Jira Service Desk[42]	Proporciona un espacio de trabajo completo para los equipos de soporte, incluyendo portal de clientes, gestión de problemas y solicitudes, e informes.	Comercial
OTRS[43]	Optimiza la prestación de servicios, la comunicación y los flujos de trabajo.	Código abierto
ManageEngine ServiceDesk[44]	Gestiona las solicitudes e incidentes de los clientes. Centraliza los puntos de contacto de la empresa para promover una comunicación eficiente.	Comercial

Por profundizar más en una herramienta, **ManageEngine ServiceDesk** (a partir de ahora, **SD+**) es una solución para la gestión y administración de servicios perfecta para una organización.

SD+ dispone de una base de datos de la gestión de configuración o CMDB[45], esta es una de las partes más importantes, puesto que ahí se almacena toda la información referente a cada cliente. En la CMDB se puede filtrar por cuenta, de esta forma se mostrarán solo los datos de esa cuenta o cliente, puesto que cada cuenta tiene un cliente asociado con su nombre. También se puede filtrar por tipo de activo, ya sean servidores, *routers*, impresoras... Como se ve en la figura 5.1, en la CMDB se están visualizando de todas las cuentas disponibles, únicamente servidores. En la imagen se puede apreciar que cada servidor tiene un nombre, modelo, sistema operativo, dirección IP, además del estado del activo indicando si está en uso o está fuera de servicio.

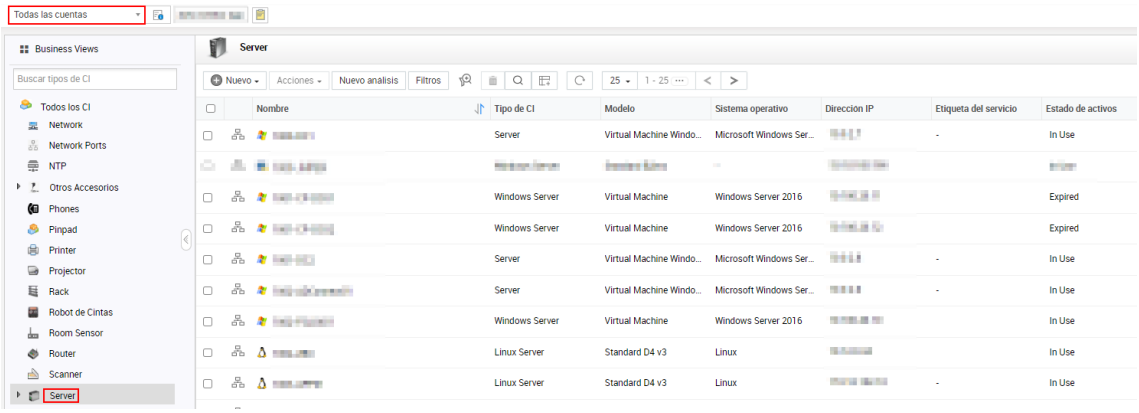


Figura 5.1: Servidores mostrados en la CMDB de la herramienta **SD+**

[40] https://www.freshworks.com/es/freshdesk/.
[41] https://osticket.com/.
[42] https://www.atlassian.com/software/jira/service-management/features/service-desk.
[43] https://otrs.com/es/home/.
[44] https://www.manageengine.com/es/service-desk/.
[45] Del inglés *Configuration Management Data Base*.

El apartado de cambios de la herramienta **SD+** se utiliza para tener un proceso detallado donde se solicita, programa y aprueba el cambio de configuración que se va a realizar, además de servir como seguimiento. De esta forma se logra mantener la estabilidad y continuidad del servicio. En este caso, los cambios se crearán para tener un seguimiento pormenorizado del parcheo de servidores. Para que un cambio se pueda llevar a cabo, se necesita la aprobación de un Comité Asesor de Cambios (CAB)[46]. Este comité es el encargado de valorar si el cambio ha de implementarse, además de aportar alguna recomendación o corrección si fuera necesaria.

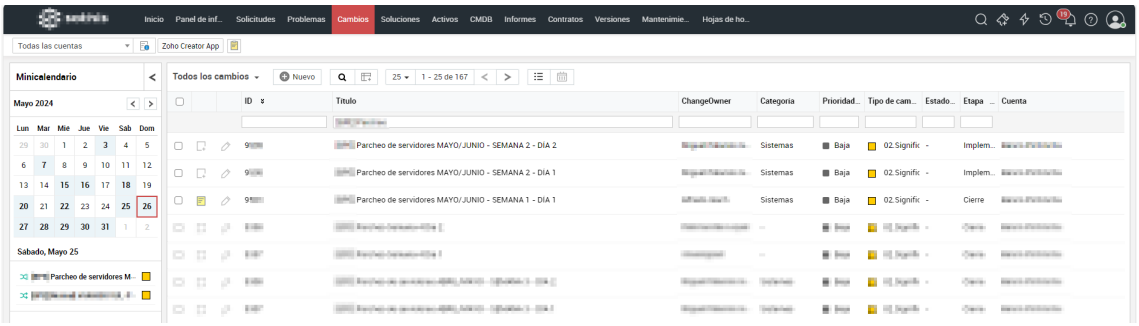


Figura 5.2: Cambios preparados para el parcheo de servidores de la herramienta **SD+**

En la figura 5.2 se muestra un calendario donde aparecen sombreados los días que hay algún cambio planificado. Además, al seleccionar un día, como se observa con el sábado veinticinco de mayo, hay dos cambios asociados para realizarse durante ese día. También se refleja en la imagen el identificador del cambio, el título, el propietario del cambio, que prioridad tiene, que tipo de cambio es, la etapa en la que se encuentra y la cuenta (o cliente) al que pertenece.

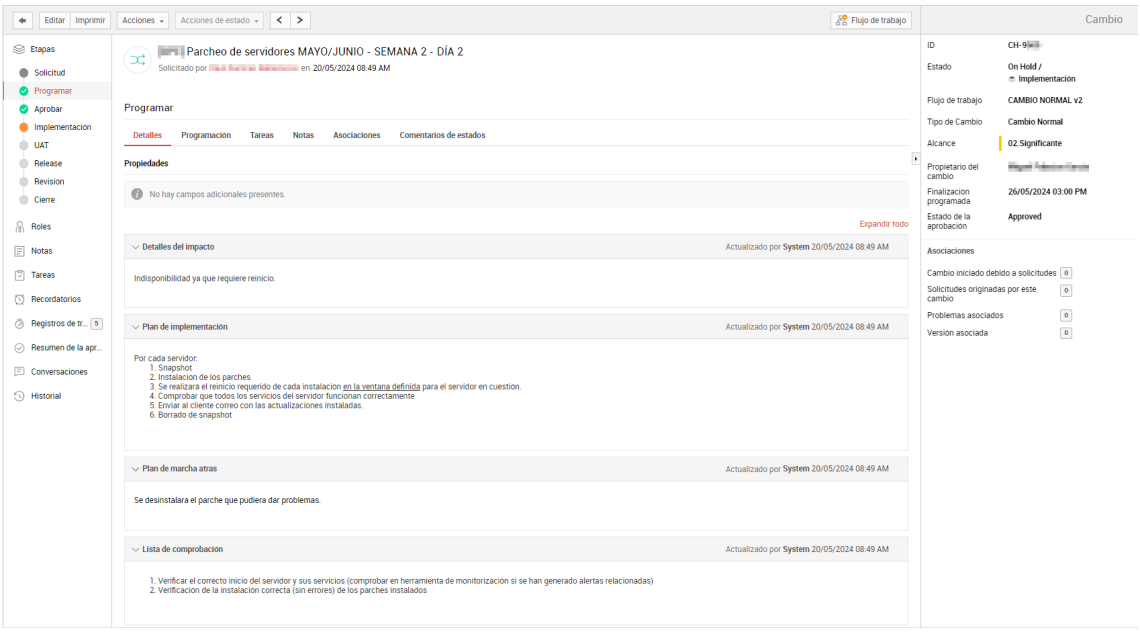


Figura 5.3: Programación de un cambio de la herramienta **SD+**

Cuando se crea un cambio, se debe crear una solicitud y luego programarlo, donde se deberá especificar: el impacto que va a tener en el servidor, cuál va a ser el plan a implementar, detallar el plan de marcha atrás existente y qué comprobaciones se van a

[46] Del inglés *Change Advisory Board*.

realizar posteriormente a la ejecución del cambio. Todas estas características se muestran en la figura 5.3.

Si se quisiera profundizar más sobre las posibles configuraciones que ofrece **SD+**, se puede consultar su página web mencionada en la tabla 5.1.

5.2 Herramientas de administración y gestión de conexiones remotas

Las herramientas de administración y gestión de conexiones remotas permiten acceder y controlar dispositivos y sistemas desde ubicaciones remotas. Estas herramientas son esenciales para la resolución de problemas, la administración de servidores y la asistencia técnica sin necesidad de presencia física.

En la tabla 5.2 se listan diversas herramientas actuales de administración y gestión de conexiones remotas.

Tabla 5.2: Comparación herramientas de conexiones remotas

Herramienta	Descripción	Características únicas
Splashtop[47]	Solución segura y rápida para el acceso remoto a equipos.	Soporta varios protocolos y tipos de conexión remota.
ManageEngine Remote Access Plus[48]	Facilita la administración y el control de los recursos de red.	Ofrece diversas herramientas como el uso compartido de escritorio remoto avanzado, la línea de comandos remotos, el registro, la activación en LAN, el apagado remoto, y el administrador de archivos.
Dameware[49]	Permite solucionar problemas de usuarios finales desde cualquier lugar.	Permite integrar soluciones de terceros como Slack, Microsoft Dynamics 365 y Salesforce.
TeamViewer[50]	Permite a los usuarios conectarse a múltiples estaciones de trabajo de forma remota.	Ofrece gestión de ordenadores y contactos, descubrimiento automático y controles integrados de supervisión, así como gestión de usuarios y dispositivos.
Devolutions Remote Desktop Manager [51]	Centralización de las conexiones remotas en una plataforma compartida. Almacena todas las contraseñas y credenciales de manera segura.	Ofrece funcionalidades de transferencia segura de archivos, informes, acceso sin vigilancia, funcionalidad de pantalla negra que bloquea las entradas de teclado y ratón en las máquinas remotas.

Por profundizar más en una de ellas, **Devolutions Remote Desktop Manager** (a partir de ahora **RDM**) es una herramienta que va a ser de gran utilidad para poder acceder de una forma segura y eficiente a los servidores que se quieran parchear.

[47] https://www.splashtop.com/es.
[48] https://www.manageengine.com/es/remote-desktop-management/.
[49] https://www.solarwinds.com/dameware.
[50] https://www.teamviewer.com/es/.
[51] https://devolutions.net/remote-desktop-manager/.

En el gestor **RDM** se pueden organizar los accesos remotos en carpetas. Esta utilidad es importante para separar los accesos por los distintos equipos que van a acceder, dado que probablemente usarán credenciales distintas, y además, separar los accesos por cliente. En la imagen 5.4 se puede observar, como está el panel separado por equipos, siendo uno de ellos el de operaciones y dentro se encuentran las carpetas organizadas por clientes.

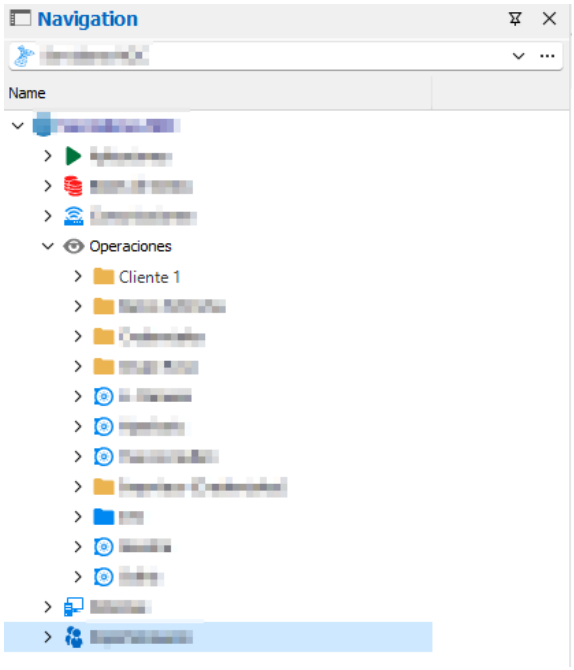


Figura 5.4: Panel de navegación de **RDM**

Con esta herramienta también se pueden agrupar las credenciales de acceso para cada servidor por clientes o según se desee. Este listado de credenciales se puede observar en la figura 5.5.

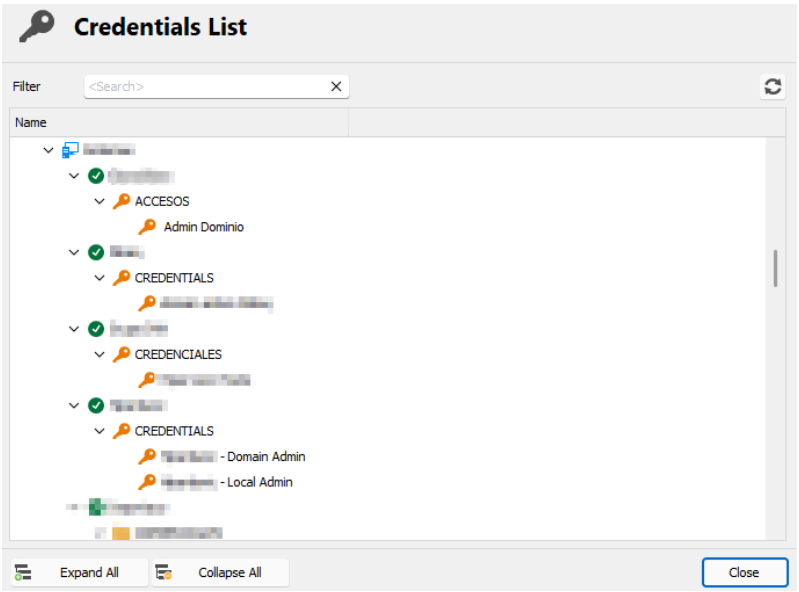


Figura 5.5: Listado de credenciales de **RDM**.

En cada servidor, para optimizar el tiempo, se puede dejar almacenada la contraseña y el usuario como se refleja en la imagen 5.6. También se pueden configurar otras características como el nombre, carpeta donde se almacena o si preguntar siempre la contraseña para entrar o no.

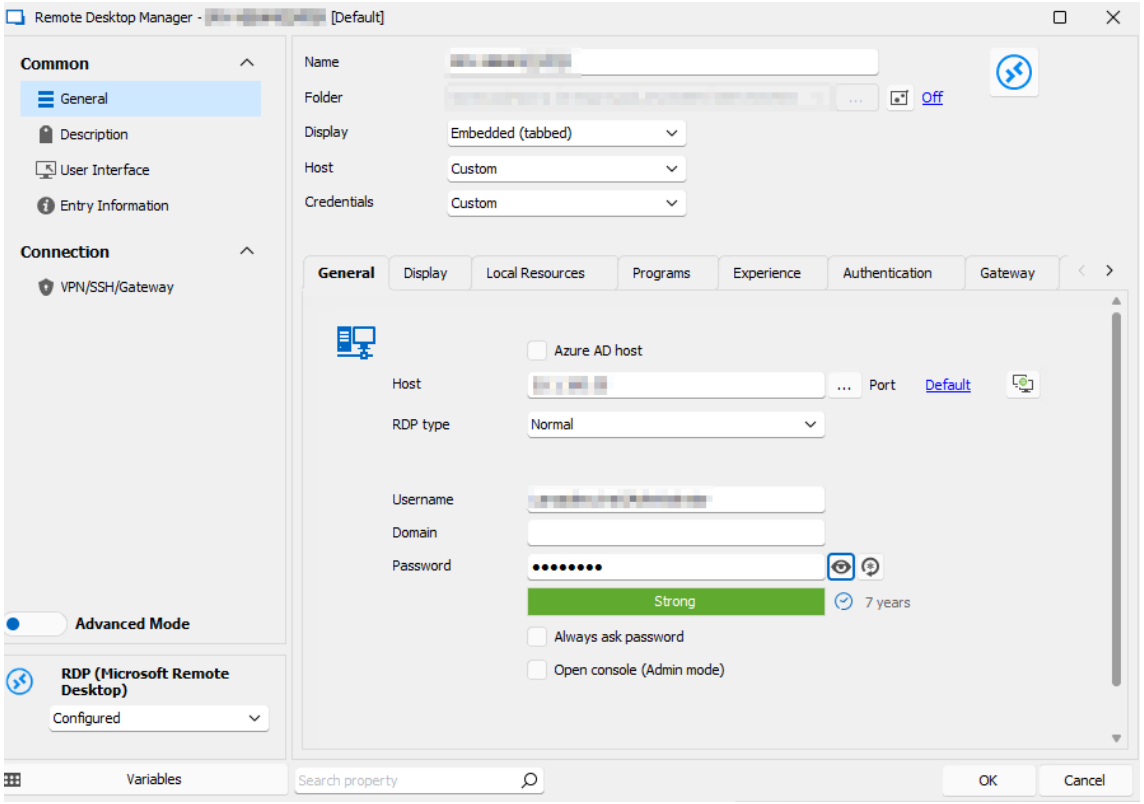


Figura 5.6: Opciones de configuración para el acceso a un servidor en **RDM**.

Al clicar una vez sobre el servidor que se quiere acceder, si se va al apartado de *Permissions*, como se muestra en la figura 5.7, se pueden observar los permisos que se tienen con el usuario actual y las modificaciones que se pueden realizar sobre ese servidor desde el **RDM**.

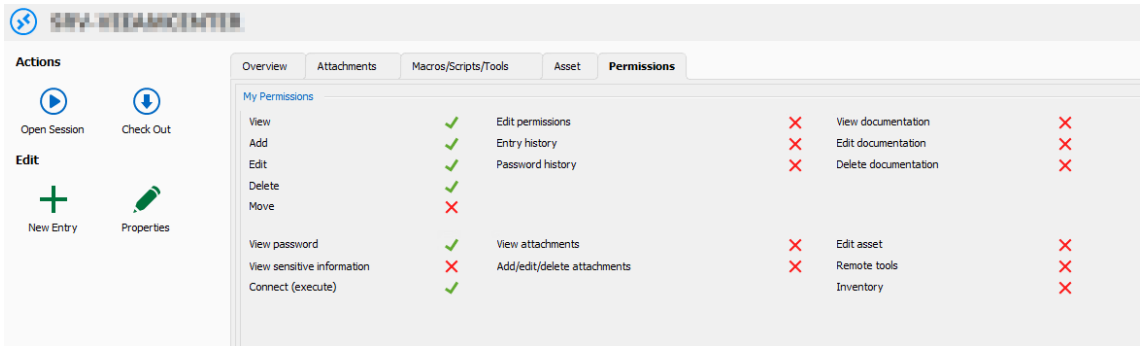


Figura 5.7: Permisos de acceso a un servidor en **RDM**.

Si se quisiera profundizar más sobre las posibles configuraciones que ofrece **Devolutions Remote Desktop Manager**, se puede consultar su página web mencionada en la tabla 5.2.

5.3 Herramientas de monitorización

La monitorización de activos es una tarea clave en todas las organizaciones, desde el monitoreo de elementos de red como *switches* o *firewalls* hasta la monitorización de servidores o cabinas de almacenamiento.

Con estas herramientas se logra observar el rendimiento de los sistemas, y poder anticipar problemas gracias a gráficas, como por ejemplo, observando el espacio de crecimiento en disco de un servidor, su consumo de memoria... Para poder actuar antes que estos problemas generen un grave impacto en la organización.

También son de gran ayuda para la pronta detección y recuperación de fallos, gracias a las alertas que proporcionan, consiguiendo así avisar a los técnicos adecuados e intervenir en el menor tiempo posible.

En la tabla 5.3 se pueden observar distintas herramientas de monitorización.

Tabla 5.3: Comparación herramientas de monitorización

Herramienta	**Descripción**	**Código abierto**
ManageEngine OpManager[52]	Ofrece una visión proactiva del estado de los servidores con un panel único para cada servidor ESX.	No
Site24x7[53]	Permite programar tareas de mantenimiento e informes personalizados para mantener los servidores en óptimas condiciones.	No
Server and Application Monitor[54]	Proporciona una interfaz web integrada y un mapeo inteligente de las dependencias de la infraestructura de las aplicaciones.	No
Better Stack[55]	Ofrece una interfaz integrada, paneles de control y cola activa, y alertas procesables ilimitadas.	No
Zabbix[56]	Es un sistema de monitoreo de redes de código abierto que permite monitorizar y registrar el estado de varios servicios de red, servidores y hardware de red.	Sí

Por profundizar más en una de ellas, **Zabbix** es una herramienta de muchísima utilidad, ya que permite recopilar datos de los activos en cuestión, normalmente mediante el protocolo simple de administración de red o SNMP. Asimismo, se pueden establecer los intervalos de tiempo deseados de los que se quiere obtener datos, además de gráficos en tiempo real. Un factor clave de **Zabbix** son las alertas altamente configurables, permitiendo adaptar el envío de notificaciones y que se ejecuten acciones automáticas.

Con **Zabbix** se pueden configurar los parámetros que se quieren monitorizar. Como se observa en la figura 5.8, se puede monitorizar el tiempo que el servidor lleva activo, la versión del agente instalado, el porcentaje de carga del procesador y su uso de memoria en el momento actual.

[52] https://www.manageengine.com/es/network-monitoring/.
[53] https://www.site24x7.com/es/.
[54] https://www.solarwinds.com/server-application-monitor.
[55] https://betterstack.com/.
[56] https://www.zabbix.com/.

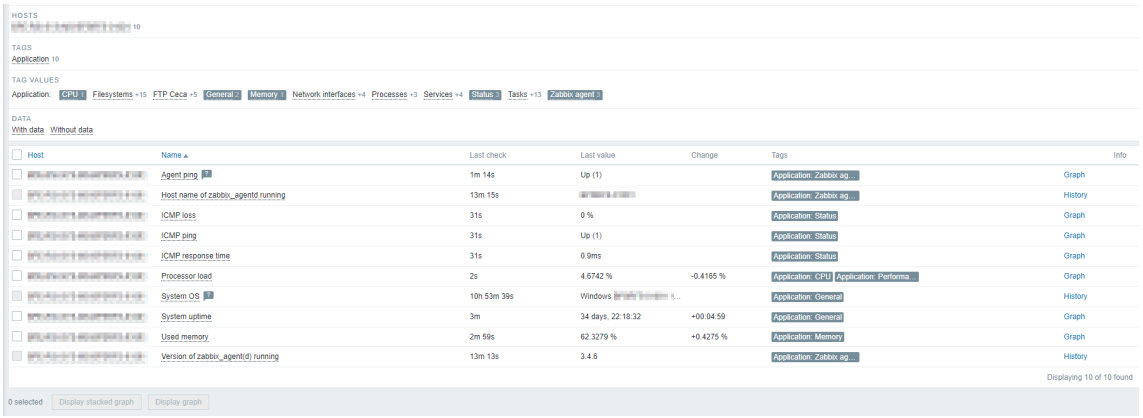


Figura 5.8: Características recopiladas de un servidor mediante **Zabbix**

Además de obtener esa información en texto, **Zabbix** también permite el uso de gráficas para ver datos como porcentajes de uso de cpu, memoria o tráfico de red. En estas gráficas se puede configurar el intervalo de tiempo a visualizar. En la imagen 5.9 se observa una gráfica del porcentaje de uso de cpu del procesador de un servidor durante los últimos treinta días. También se puede observar en la imagen el *trigger* o desencadenante de una alerta. Esto se usa para que cuando la carga de cpu del procesador superase el 90 %, se creara una alerta para avisar a los equipos necesarios para su revisión.

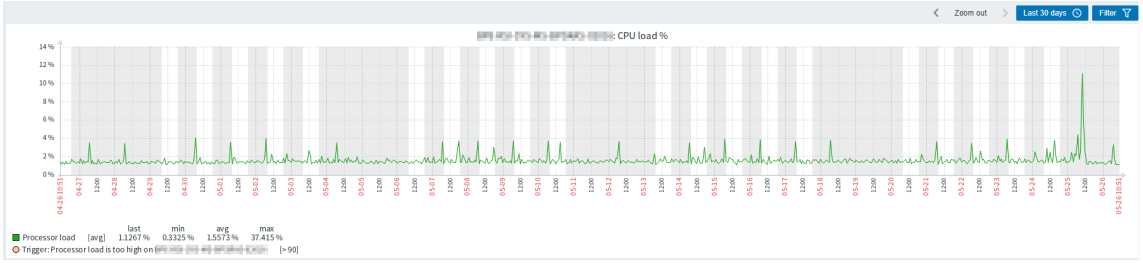


Figura 5.9: Gráfica del uso de CPU de un servidor mediante **Zabbix**

Para ver las alertas o problemas que se han configurado para cada servidor, se podrá ver desde el apartado de **Problems**. En esta página se verán todas las alertas que pueda haber, con sus distintas criticidades. Como se observa en la figura 5.10 se puede ver el histórico de alertas, donde se muestran las dos últimas que ya han sido resueltas y reconocidas (ack)[57] por el equipo de monitorización. En ese reconocimiento, se debe buscar si esas alertas están asociadas a alguna intervención, como un cambio para parchear un servidor y añadirla al cambio. En caso negativo, se debe avisar al equipo indicado para que revisen el motivo de la alerta. Estas alertas también se pueden filtrar por fecha, severidad, grupo de equipos o de un único activo si se quisiera.

En la imagen 5.11 se observa una alerta de reinicio de un servidor. Esta alerta acaba de aparecer debido a la duración que tiene y todavía no ha sido reconocida por el equipo de monitorización. También se puede apreciar la severidad de esta alerta, siendo todos estos apartados configurables por la organización pudiendo ser desde alertas críticas de desastre, alertas importantes, avisos, hasta alertas solo informativas.

Si se quisiera profundizar más en la multitud de configuraciones y opciones que dispone **Zabbix**, se puede consultar la documentación y el manual de su página web. [58]

[57] Abreviatura del inglés *acknowledge*.
[58] https://www.zabbix.com/documentation/current/es/manual/.

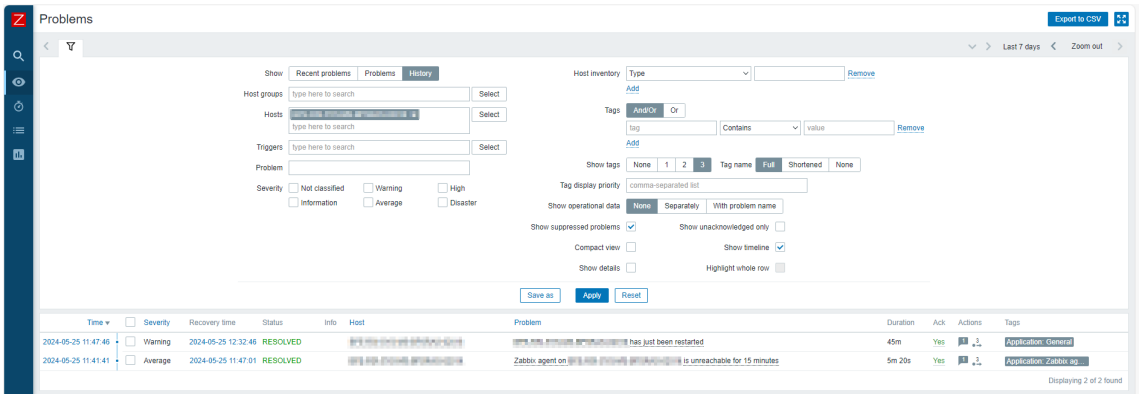


Figura 5.10: Apartado de problemas de **Zabbix**

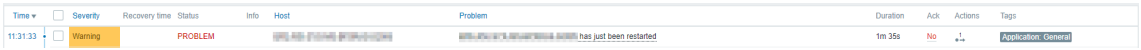


Figura 5.11: Alerta de reinicio de un servidor en **Zabbix**

5.4 Herramientas de administración y gestión de parches

En este apartado se van a describir distintas herramientas para la gestión y administración de parches y mitigación de vulnerabilidades. Estas herramientas van a lograr automatizar el proceso del parcheo de servidores, gracias a realizar esa gestión desde un único punto. Estas herramientas pueden clasificarse en función de cómo se realiza la interacción entre el servidor y la aplicación. Podemos diferenciar en herramienta con y sin agente.

5.4.1. Con agente

Estas herramientas requieren la instalación de un software o *agente* en cada servidor que se quiera parchear. De esta forma, el agente es el encargado de ejecutar en el servidor, de forma local, las órdenes y tareas mandadas desde la herramienta de gestión central y posteriormente comunicar los resultados a la aplicación central. Las principales virtudes de este planteamiento son la capacidad de controlar y gestionar sistemas, aunque estén desconectados de la red o protegidos con otras medidas de seguridad. Sin embargo, su mayor inconveniente es la necesidad de la instalación del agente en cada equipo, además de tener que mantenerlo actualizado para su correcto funcionamiento.

- **ManageEngine Endpoint Central**: Es una herramienta de gestión y administración de servidores, ordenadores, dispositivos móviles... **Endpoint Central** ofrece una gestión unificada de terminales o UEM[59] para controlar toda la organización desde un único sitio. Con esta solución se pueden administrar todos los activos de forma remota, además de poder desplegar software mediante plantillas para todos los equipos que se desee de forma automática. También se pueden instalar SO en máquinas Windows junto a sus drivers y aplicaciones necesarias.

  En relación con el tema de la seguridad, **Endpoint Central** ofrece una gestión de amenazas y vulnerabilidades, donde se pueden identificar riegos reales, controlar las configuraciones de seguridad y mitigar vulnerabilidades desde el día en el que se descubren.

[59] Del inglés *Unified Endpoint Management*.

Respecto al parcheo de servidores, **Endpoint Central** permite la administración centralizada para la aplicación de parches. De esta forma se pueden aplicar tanto las actualizaciones de Windows como parches para solucionar vulnerabilidades. El parcheo de servidores desde **Endpoint Central** se puede realizar de forma semiautomática o automática.

Para instalar los parches de una forma automática hay que configurar previamente tres apartados:

1. **Grupos personalizados**: en primer lugar, se debe crear el grupo con los activos a parchear. Esto se hará desde el apartado **Admin** y se buscará **Grupos personalizados**, procediendo a añadir uno nuevo. Se podrá nombrar el grupo como se quiera y fijar si ser un grupo estático, estático único o dinámico. El primero de todos son equipos que pueden ser añadidos de forma manual o desde el AD, el segundo grupo sigue las mismas reglas, salvo que ese equipo añadido solo puede estar en ese único grupo, y por último el grupo dinámico añade los equipos en función de reglas que se pueden configurar, como juntarlos por SO, dirección IP, fabricante...

   Para el parcheo de servidores, es aconsejable un enfoque estático, ya que permite tener más controlados y organizados todos los servidores. Como se observa en la figura 5.12 se ha creado un grupo de choque usando 2 servidores para probar los parches acumulativos mensuales de **Microsoft**.

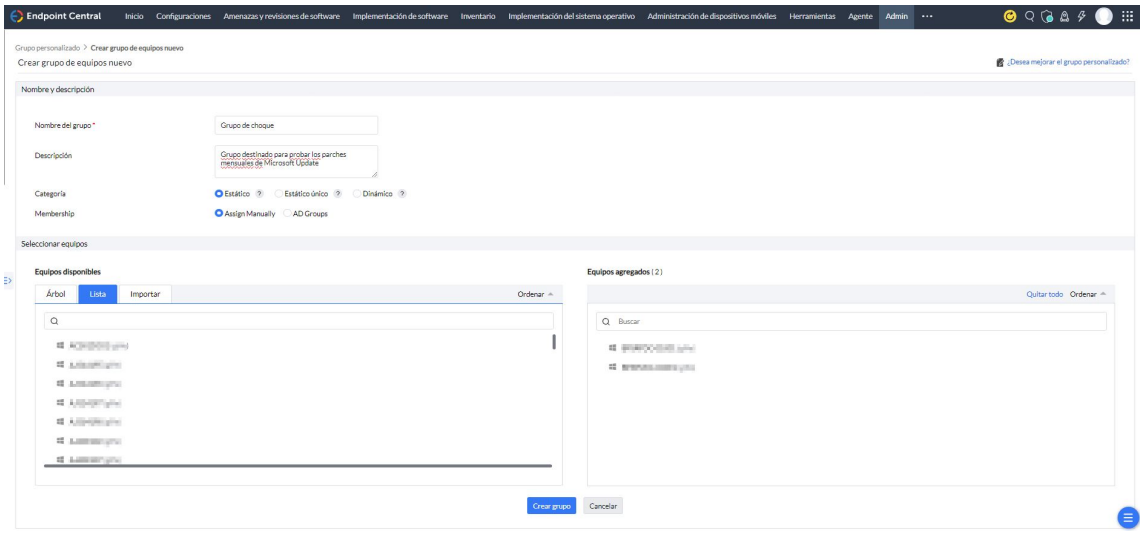


Figura 5.12: Creación de un grupo personalizado desde **Endpoint Central**

2. **Directivas de implementación**: una vez ya creados todos los grupos que queramos implementar, se va a proceder a crear las directivas de implementación, o en otras palabras, cuándo se van a implementar los parches en las máquinas. Estas directivas pueden organizarse respecto a una división regular de una semana, como se observa en la imagen 5.13, o usando la división del *Patch Tuesday*, como se contempla en la figura 5.14, concepto explicado en la sección 3.2.2. Estas dos figuras vienen incluidas al crear la directiva de implementación desde el **Endpoint Central**, para así ayudar al usuario a crearlas correctamente.

Figura 5.13: Calendario semanal acorde a la división regular de una semana

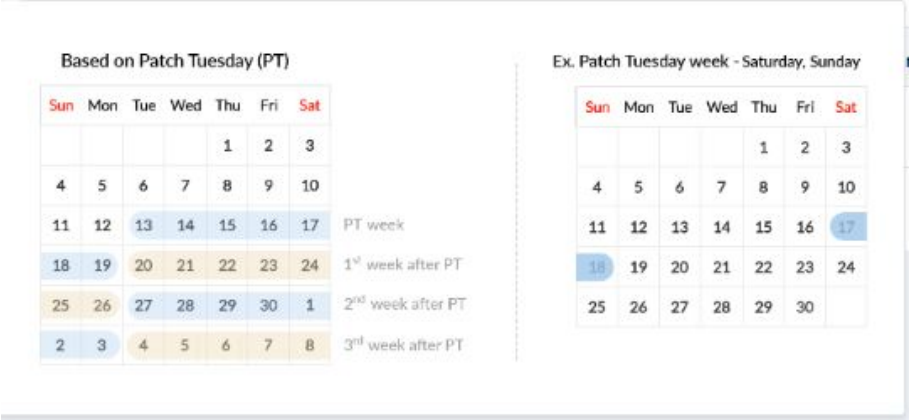


Figura 5.14: Calendario semanal acorde al *Patch Tuesday*

Una vez se ha elegido el tipo de división de semana a utilizar, se deben escoger los días en que se va a realizar esta implementación, además de en qué semana. Como se observa en la imagen 5.15, se puede ver que se ha seleccionado que los parches se apliquen la primera semana después del martes de actualización y el día seleccionado sea el sábado de siete a diez de la mañana. También se ha configurado que los parches solo se descarguen mediante el agente instalado en el servidor, durante el periodo de implementación y no durante cualquier momento.

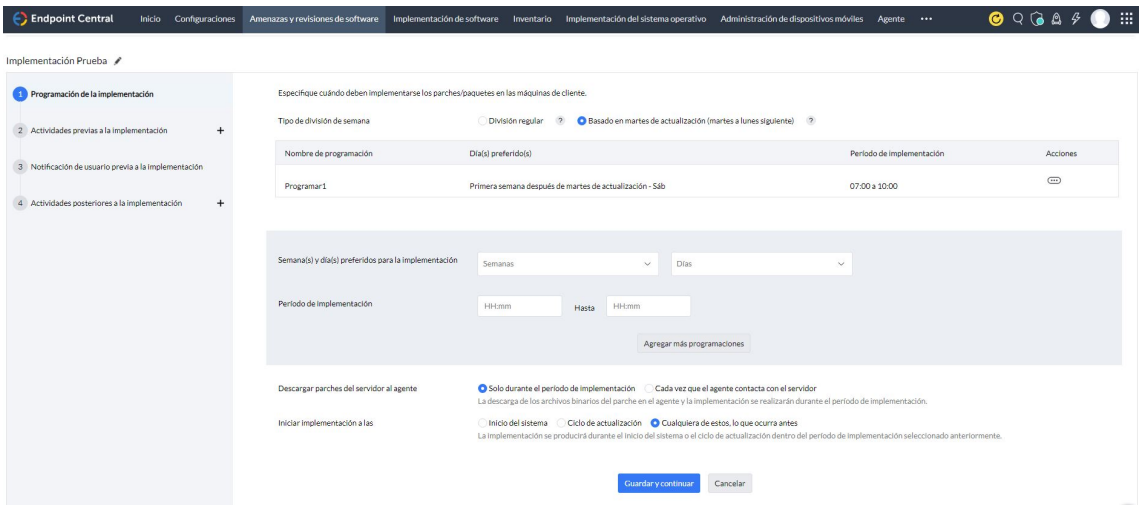


Figura 5.15: Creación de una directiva de implementación desde **Endpoint Central**

Se pueden configurar actividades previas a la implementación, como por ejemplo, provocarle un reinicio al servidor, para que luego pueda instalar los parches de una forma limpia, sin que pueda tener algún proceso enganchado que provoque una mala instalación. También se pueden configurar actividades posteriores a la implementación, como otro reinicio para que los parches puedan terminar de instalarse correctamente. Como se aprecia en la figura 5.16, se puede observar como se ha configurado para que reinicie todos los equipos si la implementación ha sido correcta, dando 5 minutos de espera por si hubiera algún usuario conectado.

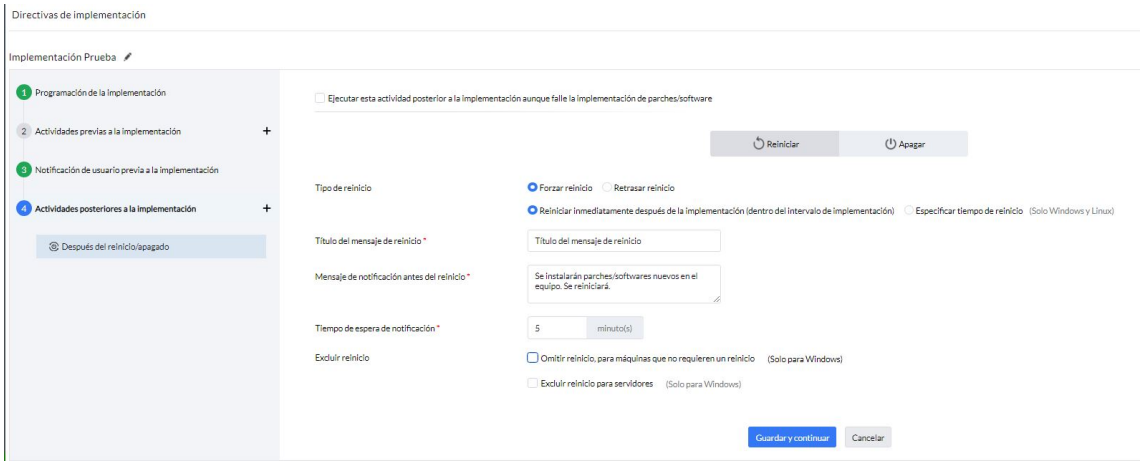


Figura 5.16: Actividades posteriores de una directiva de implementación desde **Endpoint Central**

3. **Implementación automática**: por último se debe crear la implementación automática. En este apartado se van a seleccionar los parches que se quieren instalar. Como se ve en la figura 5.17, se han seleccionado los parches acumulativos y de gravedad crítica, siendo estos todos los parches acumulativos de seguridad mensuales de Windows Update. En este caso se ha configurado para que solo se implementen para los *Windows Server 2019 Standard Edition (x64)*. Una vez se han seleccionado las actualizaciones y aplicaciones seleccionadas, se elegirá la configuración de implementación deseada, en este apartado se elegirá la directiva de implementación que se ha creado en el apartado anterior. Posteriormente, se debe elegir el destino de esta implementación automática, debiéndose elegir el grupo personalizado que se creó al principio. En el apartado de configurar notificaciones, se puede configurar para que lleguen alertas al correo del proceso o del resultado. El intervalo de tiempo para recibir estas alertas también es configurable.

   Con estos tres pasos, se acaba de configurar una implementación automática, para que cada sábado, de siete a diez de la mañana, de la primera semana después del *Patch Tuesday* se instalen en los dos servidores seleccionados el parche acumulativo mensual para el SO Windows Server 2019.

Finalmente, desde el apartado de implementación automática, si se clica sobre la tarea creada se puede ver en tiempo real el estado de ejecución de la tarea, además del resultado final. En la figura 5.18 se pueden ver 2 servidores de los cuales uno de ellos le faltan todavía parches por aplicar, mientras que el otro ha instalado correctamente los dos parches pendientes que tenía.

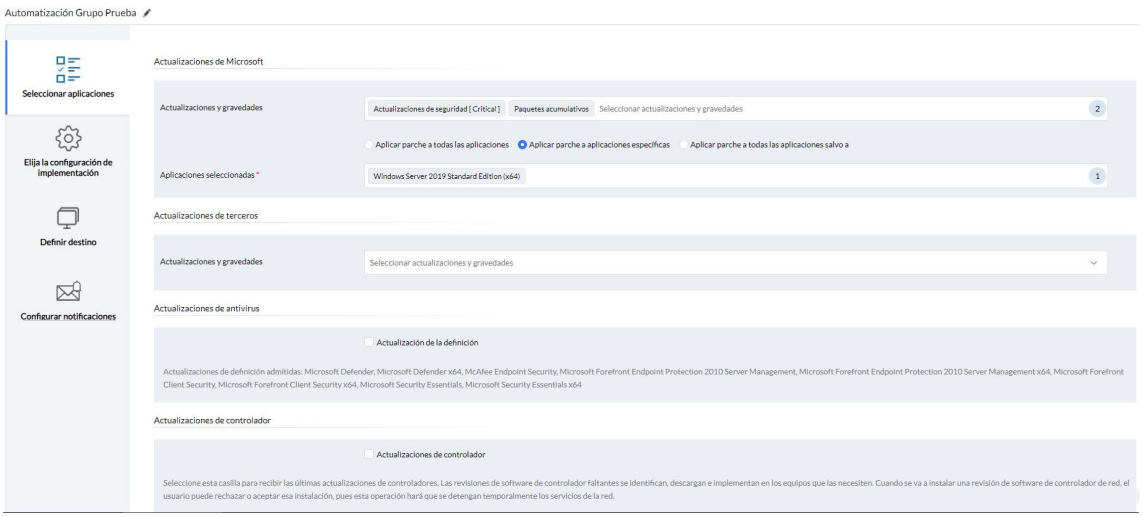


Figura 5.17: Implementación automática desde **Endpoint Central**.

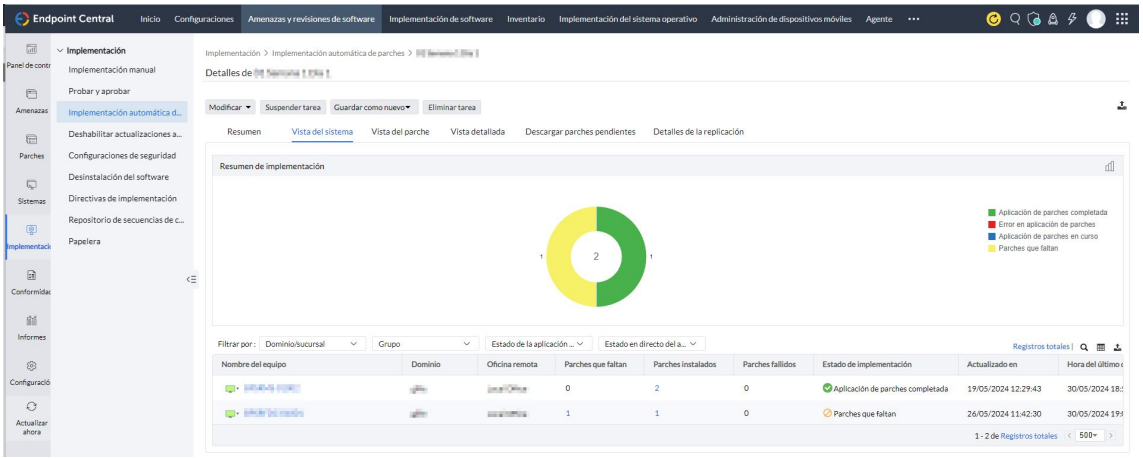


Figura 5.18: Resultado de una implementación automática desde **Endpoint Central**

Si se quisiera profundizar más en como funciona **ManageEngine Endpoint Central** para la gestión de parches, creación de implementaciones manuales, u otras acciones, se puede consultar su página web.[60]

5.4.2. Sin agente

Estas herramientas funcionan sin la necesidad de tener un software externo instalado en los servidores que se van a parchear. En cambio, las tareas a realizar se envían de forma remota a través de la red utilizando protocolos estándar. Sus principales ventajas son la necesidad de un menor mantenimiento, y una forma de implementación menos compleja. Por el contrario, estas herramientas pueden estar más limitadas al no poder conectar a servidores fuera de la red o que tengan configuradas ciertas reglas en los Firewalls que impidan el acceso u otras medidas de seguridad. A continuación se van a detallar distintas herramientas o aplicaciones.

- **WSUS**[61]: Un servidor **WSUS** es una herramienta que permite administrar y distribuir actualizaciones a través de una consola de gestión. Este servidor puede tam-

[60] https://www.manageengine.com/products/desktop-central/.
[61] *Windows Server Update Services*.

bién servir como fuente de actualizaciones para otros servidores **WSUS** dentro de la misma organización. El servidor **WSUS** que actúa como fuente de actualizaciones es el que se encuentra en la posición superior de la cadena. Para una implementación correcta, al menos debe haber un servidor **WSUS** de la red, que deba tener conexión a la página de Microsoft Update para poder comprobar actualizaciones nuevas o descargar los parches. Según la configuración de la organización, se pueden conectar más o menos servidores directamente a Microsoft Update.

En primer lugar, se deben comprobar los requisitos mínimos necesarios tanto de hardware como de software para poder habilitar el rol de servidor **WSUS**. Cuando se ha comprobado, desde el panel de **Administrador del servidor** en el apartado de agregar roles, añadiremos el rol **Windows Server Update Services**. Posteriormente, se debe elegir si se va a usar una implementación simple de **WSUS**, usando un único servidor que se conecte a Microsoft Update y para la descarga de actualizaciones. Ese contacto entre el servidor y Microsoft Update se denomina **sincronización** y en este proceso, se comprueba si desde la última sincronización ha habido algún parche nuevo para instalar. Para habilitar la conexión y poder descargar las actualizaciones desde el servidor **WSUS**, se deben habilitar en el firewall, que separa Internet de la red interna de la organización, los puertos para los protocolos de transferencia de hipertexto y protocolos seguros de transferencia de hipertexto, **HTTP**[62] y **HTTPS**[63], respectivamente, siendo el 80 y el 443, respectivamente, aunque se puedan modificar. Se puede jerarquizar la relación entre servidores **WSUS**, teniendo un servidor **WSUS** primario, que sea el que descargue las actualizaciones y las distribuya a otros servidores **WSUS** que continúen la cadena distribuyendo las actualizaciones al resto de equipos, consiguiendo un ahorro en el ancho de banda, aunque el nivel máximo de profundidad de la jerarquía recomendado por Microsoft es de tres niveles.

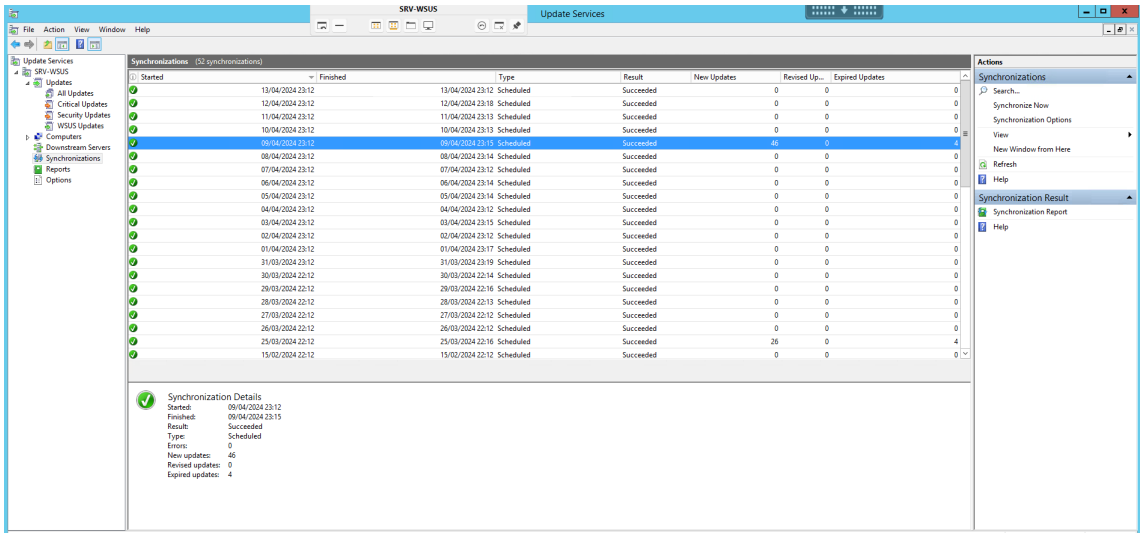


Figura 5.19: Sincronización del servidor **WSUS** con Microsoft Update

En la figura 5.19 se puede observar una configuración para realizar la sincronización, buscando actualizaciones críticas cada día, siendo el segundo martes de cada mes (*Patch Tuesday*) cuando aparecen actualizaciones nuevas.

Se pueden agrupar los servidores como se explicó en el punto 3.1.4 para que reciban siempre las mismas actualizaciones y en el momento deseado. En caso de que

[62] Del inglés *Hypertext Transfer Protocol*.
[63] Del inglés *Hypertext Transfer Protocol Secure*.

un servidor esté en varios grupos y cada grupo tenga alguna acción que realizar, tendrá mayor prioridad el grupo que tenga mayor profundidad en una rama. A modo de ejemplo, consideremos una jerarquía de agrupación de servidores como la siguiente:

- Servidores
  - ServidoresVLC
    - Producción
    - Pruebas
  - ServidoresMAD
    - Desarrollo
- Equipos de escritorio
  - EquiposChoque

Como se puede observar en este ejemplo, el grupo *Pruebas* tiene mayor prioridad que el grupo *EquiposChoque*, por lo que cualquier sistema que se encuentre en ambos grupos, tendrá prioridad una acción del grupo *Pruebas*.

Las actualizaciones pueden ser filtradas por producto deseado, por criticidad o por idioma y para que se puedan aplicar a los equipos deseados deben ser previamente aprobadas. Estas aprobaciones se pueden realizar de forma manual, seleccionando el parche que se quiere aplicar, o se pueden crear reglas para que las actualizaciones de una clasificación específica o de una criticidad en concreto se aprueben de forma automática. Posteriormente, se pueden revisar los informes que se han generado, comprobando el estado de las actualizaciones, para ver si están instaladas correctamente o ha habido algún problema.

Por último, si se dispone de un entorno con *Active Directory*, los equipos pueden obtener de forma automática las actualizaciones con **WSUS**, mediante las directivas de grupo (GPO). Se pueden configurar diversas extensiones para fijar la forma de interacción entre los equipos de dominio y que sean clientes de **WSUS**. Se puede establecer la frecuencia con la que sincronice Windows para buscar actualizaciones automáticas.

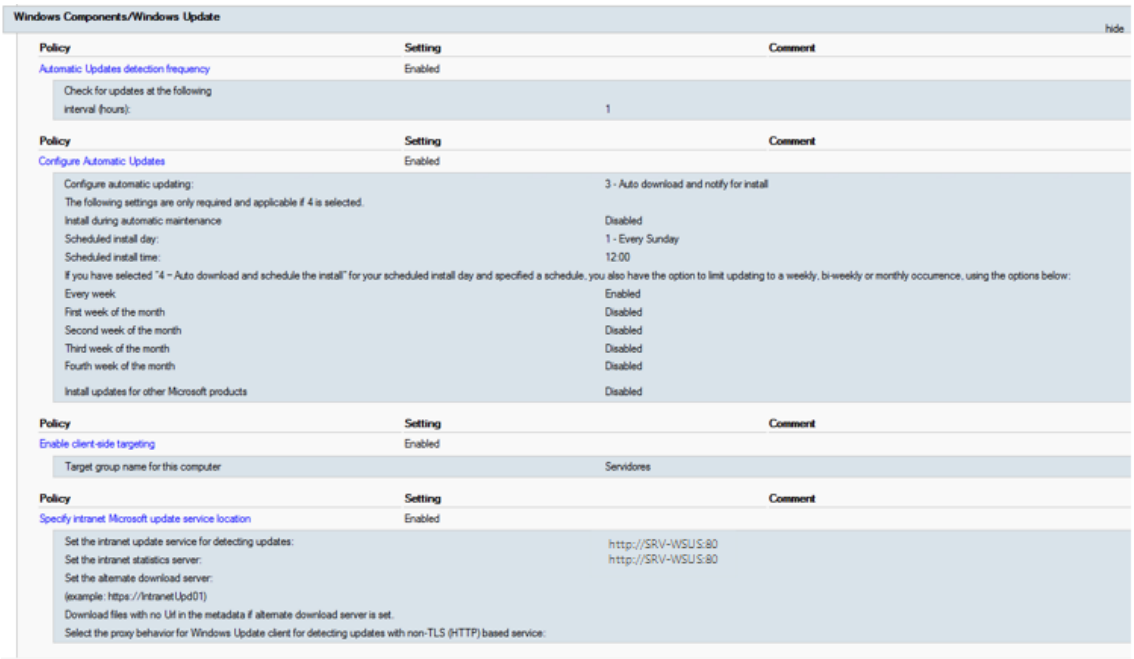


Figura 5.20: GPO configurada para la descarga e instalación de los parches al grupo *Servidores*

En la figura 5.20, se puede observar una GPO configurada para la descarga e instalación de actualizaciones, para que se instale cada domingo a las doce de la mañana en todos los servidores que se hayan añadido al grupo **WSUS** de *Servidores*. También se puede observar cómo es el servidor *SRV-WSUS*, a través del puerto 80, el que se conecta mediante HTTP al catálogo de Microsoft Update.

Si se quisiera profundizar más en la explicación de cómo configurar un servidor **WSUS** o el gran abanico de posibilidades que ofrece en cuanto a configuraciones, se debe consultar la página web de **Microsoft Learn** [27].

- **Batchpatch**: es una herramienta para la administración de parches de Windows. **Batchpatch** instala los parches de forma remota mediante la red. Con **Batchpatch** se pueden aplicar múltiples parches en múltiples equipos y reiniciarlos de forma simultánea. Además, proporciona una herramienta de monitorización para observar en tiempo real el progreso de las actualizaciones y ver si el equipo responde correctamente.

  Para usar **Batchpatch** sin limitaciones se debe acceder a la consola de gestión de **Batchpatch** con elevación de privilegios. También se requiere que se permita la comunicación entre **Batchpatch** y los servidores de destino de las actualizaciones, por lo que se debe configurar correctamente los firewalls para permitir la comunicación y la administración remota.

  Batchpatch dispone de una interfaz simple, sencilla e intuitiva. En la imagen 5.21 se muestra la interfaz básica al abrir **Batchpatch** por primera vez.

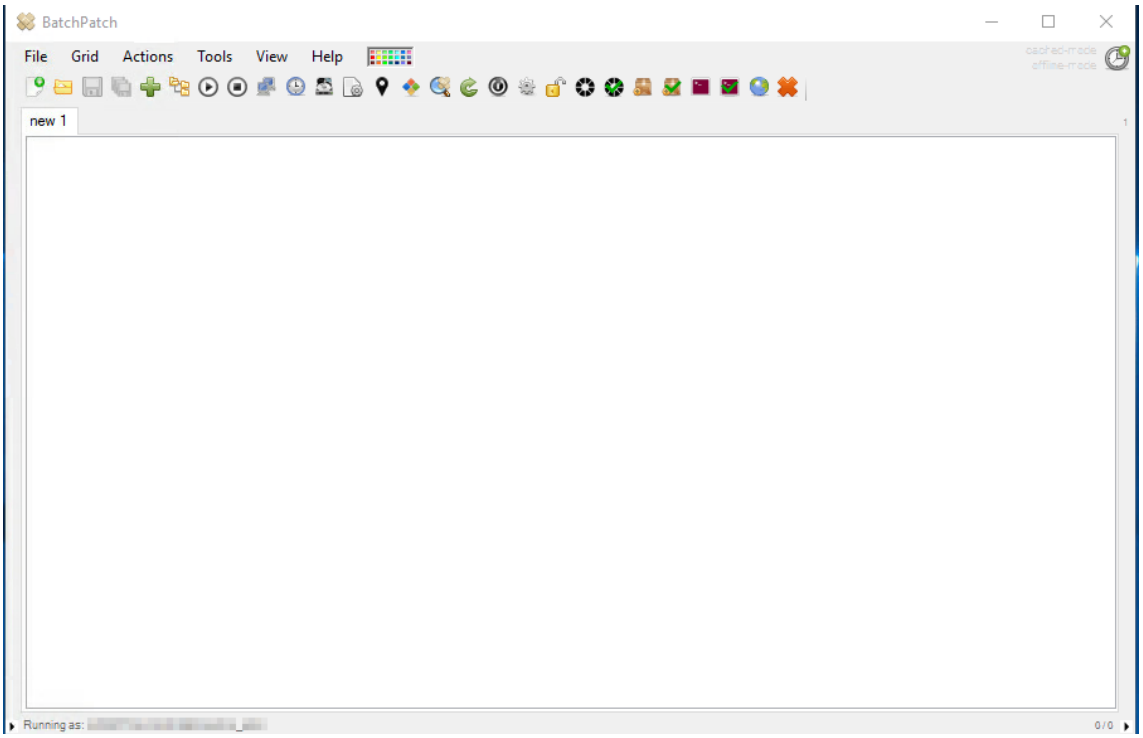


Figura 5.21: Interfaz de **Batchpatch**

Desde el apartado de **Grid** se pueden seleccionar los servidores que se van a querer actualizar. Se pueden añadir manualmente escribiendo la lista de equipos, por nombre o por IP, y pegarla en la ventana de *Add hosts*. También se pueden importar los nombres de los equipos desde el AD mediante la opción *Add hosts from directory*. Estas opciones se muestran en la figura 5.22. El listado de servidores también se puede añadir desde **File** e importando un archivo ya con la lista creada. Se pueden

crear varias pestañas y tener los servidores en varias pestañas para una mejor organización, separándolos, por ejemplo, por días u horas en función del volumen de servidores a parchear.

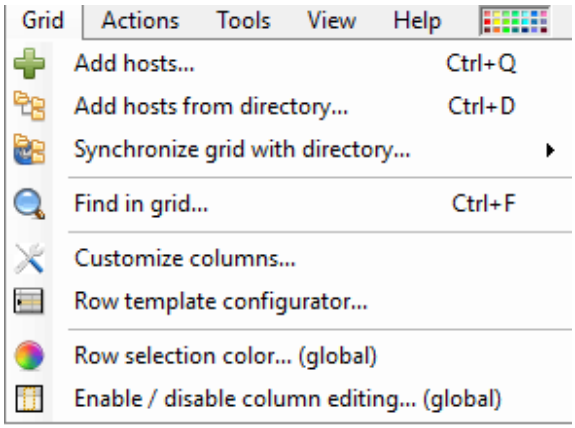


Figura 5.22: Añadir servidores en **Batchpatch**

Cuando los *hosts*[64] ya están añadidos, si se quiere obtener más información de uno o varios de ellos se seleccionan dándole un clic y se sombreará la celda de color amarillo. Entonces, desde el apartado **Actions** de **Batchpatch** se les podrá enviar un paquete de tipo ICMP[65] y esperar recibir una respuesta *ICMP echo reply* para determinar si existe una ruta entre ambos servidores y poder medir diferentes parámetros de conexión. Esto se hará desde el apartado *Ping*.

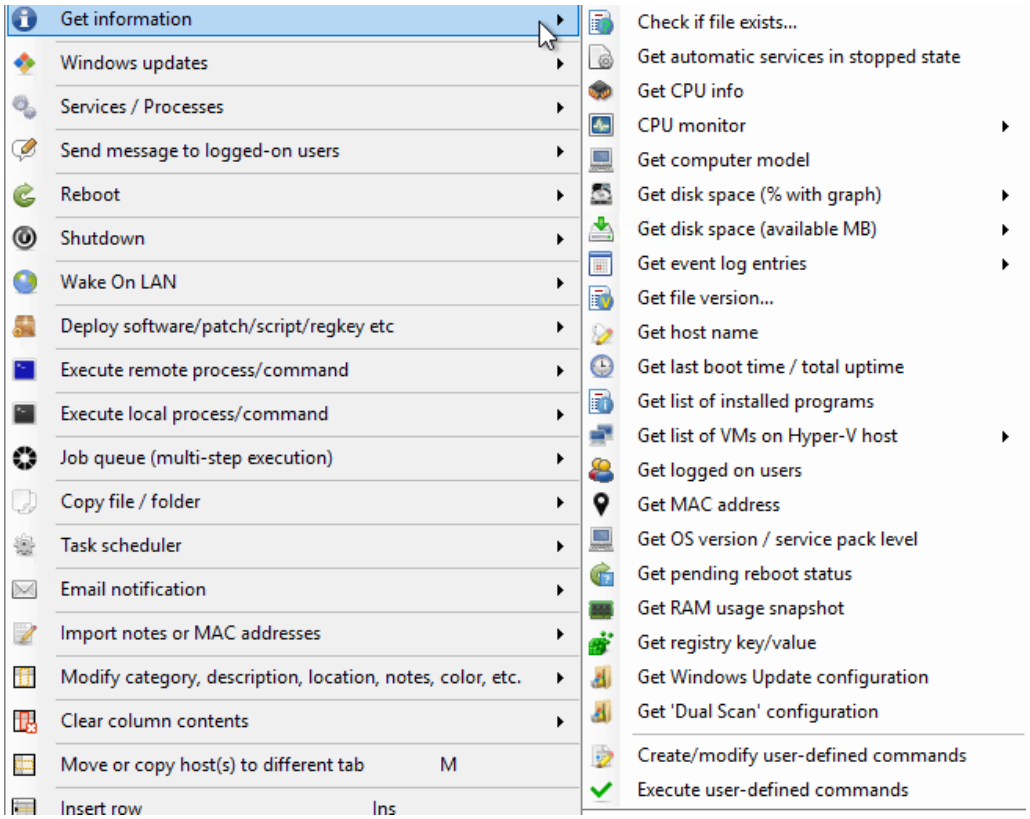


Figura 5.23: Obtener más información de un servidor en **Batchpatch**

[64] Término utilizado para referirse a equipos.
[65] Del inglés *Internet Control Message Protocol*.

También, desde el apartado *Get information* se podrán realizar varias comprobaciones que serán de utilidad antes del parcheo. Para asegurarse de que el parche se pueda instalar correctamente, se podrá comprobar el espacio en disco disponible, la versión del sistema operativo actual, el tiempo que lleva encendido el servidor... Todas estas características pueden ser comprobadas como se muestran en la figura 5.23.

Si se quiere realizar la instalación del parche de forma semiautomática, se pueden configurar manualmente los parches a ejecutar por **Batchpatch**. Desde el apartado de **Actions**, desplegando el apartado de *Deploy*[66] *software/patch/script/regkey etc* se pueden crear configuraciones. Esto se muestra en la figura 5.24.

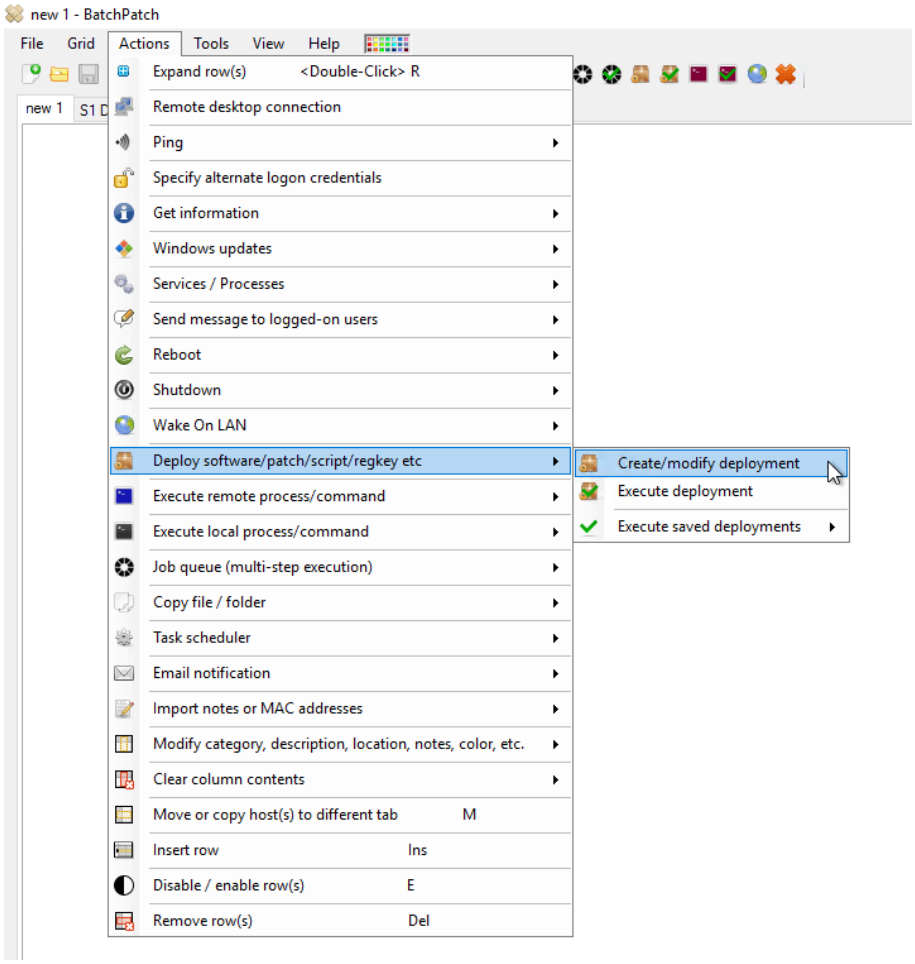


Figura 5.24: Crear una nueva implementación desde el apartado de acciones en **Batchpatch**

Creando estas configuraciones se podrán elegir los parches y scripts que van a ser aplicados a los equipos. Para crear una configuración para un parche del catálogo de Microsoft, se debe descargar y dejarlo en una carpeta donde se haya configurado el servidor con **Batchpatch**. Posteriormente, se debe poner un título a la configuración y es importante remarcar la opción de */norestart* como se ve en la figura 5.25, ya que si no se marca esa casilla, nada más instalar el parche se reiniciará el servidor, pudiendo realizarse en un momento que no sea conveniente. Para que la configuración se quede guardada en **Batchpatch** se debe clicar sobre a las dos flechas **>>** y pulsar en *Close*. Si se quisiera ejecutar ya el parche o script se podría clicar en *Execute now* o a *Apply deployment to row(s) without executing* para aplicarlo a los ser-

[66] Traducción del inglés: desplegar o implementar.

vidores que estén seleccionados. También se puede observar en la figura 5.25 otras configuraciones ya guardadas en **Batchpatch**, como comprobaciones de servicios y parches para otros sistemas operativos.

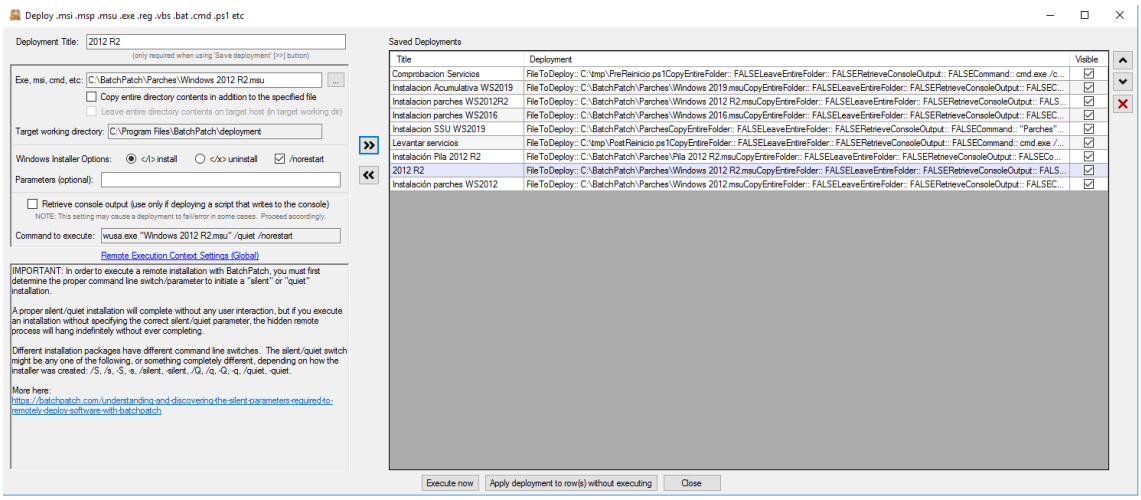


Figura 5.25: Crear una implementación para un Windows Server 2012R2

Una vez ya se tienen las configuraciones creadas, ahora se debe crear una tarea programada para que se ejecuten esas configuraciones. En este punto los servidores se pueden planificar de uno en uno o juntar los que se van a realizar a la misma hora con la misma configuración. Esto se realizará desde el apartado de **Actions** y desplegaremos la pestaña de *Task scheduler* y se le clicará a crear una nueva tarea. Existen dos opciones: crear una única tarea programada o crear múltiples tareas programadas. Para este caso, se recomienda la tarea múltiple, puesto que se pueden ejecutar comprobaciones previas y posteriores. En la imagen 5.26 se tiene un ejemplo de una tarea múltiple programada para parchear un Windows Server 2016. En primer lugar, se comprueban los servicios, y diez minutos después se instala el parche. Con dos horas de margen para que se haya completado correctamente la instalación del parche, se reinicia el servidor y una hora después se comprueba que todos los servicios se han iniciado correctamente. Estas configuraciones y horas se tendrán que adaptar a la ventana de mantenimiento proporcionada por cada cliente.

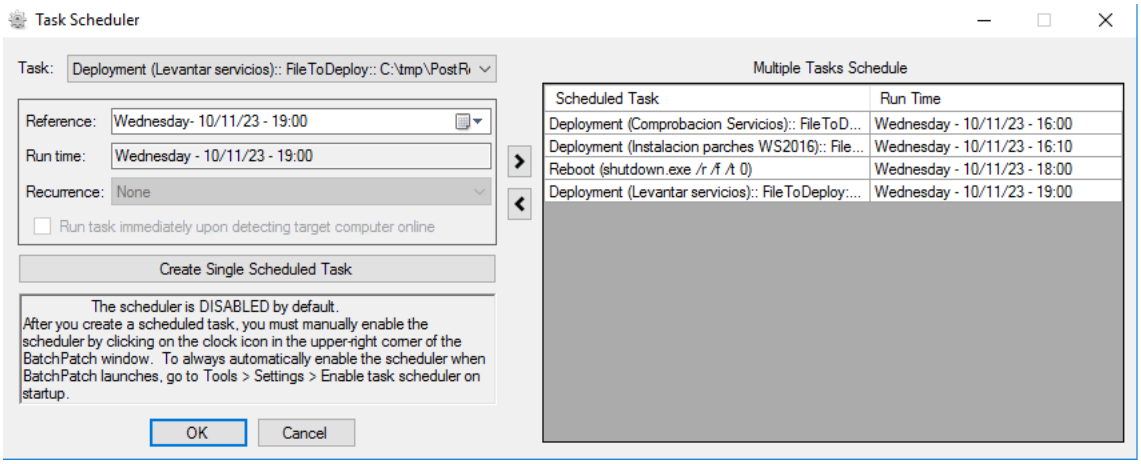


Figura 5.26: Crear una tarea programada en **Batchpatch**

Para que **Batchpatch** pueda ejecutar correctamente las tareas creadas hay dos opciones: la primera es mantener el programa de **Batchpatch** abierto, aunque el entorno de virtualización que se use se cierre, el servidor con el **Batchpatch** debe mantener la ventana abierta para poder ejecutarse, de lo contrario no se hará. La otra opción es desde el apartado de **Tools** se clicará sobre la opción *Run Batchpatch as a service*, esto convertirá **Batchpatch** en un servicio de Windows y se ejecutará de forma automática. Esto será de gran utilidad para parcheos programados fuera de hora donde no se requiera intervención directa. En la imagen 5.27 se observa cómo configurar como servicio el archivo *Prueba.bps* con las tareas programadas para el parcheo de varios servidores. El único inconveniente de usar **Batchpatch** como servicio es que imposibilita ver la interfaz con los servidores configurados y los logs hasta que no han acabado todas las tareas programadas, mientras que si se ejecuta de la otra forma se puede ir viendo el resultado paso a paso.

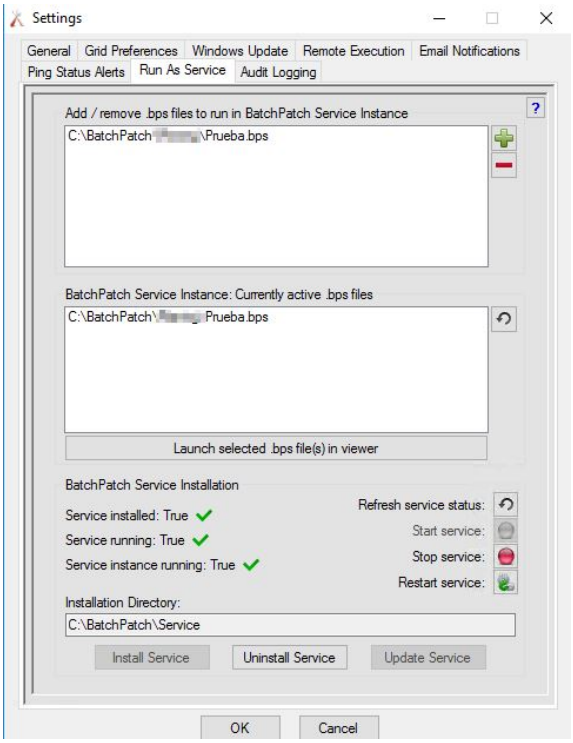


Figura 5.27: Ejecutar **Batchpatch** como servicio

Batchpatch puede combinarse con WSUS para aprobar e instalar las actualizaciones de una forma eficiente y segura. Con WSUS se puede configurar para aprobar los parches de forma automática y descargarlos en el servidor WSUS. Desde ahí, los equipos se descargarán los parches del servidor WSUS y **Batchpatch** procederá a instalar los parches ya descargados. Con esta opción se logra reducir el consumo de ancho de banda de la red, además de con WSUs recibir informes básicos sobre las actualizaciones. Según la propia página de **Batchpatch**, recomiendan el siguiente enfoque para combinar WSUS y **Batchpatch**: En primer lugar, usar GPO para que los servidores cliente se descarguen automáticamente desde el servidor WSUS, los parches ya aprobados previamente por el servidor WSUS. Posteriormente, cuando vaya a empezar la ventana de mantenimiento, desde **Batchpatch** se debe clicar sobre *Actions*, pulsar sobre *Windows Updates* y marcar la opción de *Install downloaded updates*. Si se quisiera profundizar más en las posibilidades que ofrece la herramienta **Batchpatch**, se debe consultar su página web[67].

[67] https://batchpatch.com/.

CAPÍTULO 6
Estudio de caso realista

En ese capítulo se va a realizar una simulación de cómo podría ser una interacción real con un cliente, mostrando cómo serían los primeros pasos a seguir y usando la metodología general explicada anteriormente.

6.1 Contexto previo

Bob, es presidente de una de las mayores consultoras de tecnología de los últimos tiempos, dispone de técnicos expertos en todas las materias y un trato con el cliente excelente.

Por otro lado, Alice, acaba de comprar la gran farmacéutica **FarmaLicex**, y aunque a nivel de investigación es pionera en el sector, se ha dado cuenta de que en cuanto a seguridad puede tener varias brechas importantes que debe subsanar para evitar filtraciones o posibles ataques. También observa que sus servidores no se están usando y gestionando de la forma más eficiente. Debido a todo esto Alice decide ponerse en contacto con Bob para contratar los servicios de su consultora, **BobTec Soluciones**.

6.2 Acuerdo del contrato

Tras varias propuestas, finalmente ambas partes han llegado a un acuerdo. Alice ha contratado los servicios de **BobTec Soluciones**. En el contrato se ha establecido un servicio gestionado[68] completo, por lo que dispone de los siguientes servicios:

- Monitorización continua las veinticuatro horas, los siete días de la semana (24x7).
- Equipo de ciberseguridad contratado para la mitigación de vulnerabilidades.
- Parcheo de servidores, con una periodicidad mensual.
- Informes de salud de sus sistemas y soporte técnico.

Además de estos puntos, también se ha contratado la gestión de todos sus activos de comunicaciones y de sus sistemas Linux.

[68] Es la administración y gestión de una infraestructura tecnológica por parte de terceros.

6.3 Desarrollo

Para poder realizar todas estas tareas, se necesita acceso a los servidores de la farmacéutica de Alice. Para esto, se acuerda crear un usuario nominal en el AD, y así se pueda usar en todo el dominio, para los técnicos N2, el equipo de monitorización y el equipo de operaciones, teniendo todos permisos de administrador salvo el equipo de monitorización. Para este proceso se les envía por correo electrónico el usuario con el dominio a utilizar y por SMS la contraseña para evitar brechas de seguridad. Para mayor protección, la contraseña se cambiará mensualmente.

Para otros dispositivos o aplicaciones como una cabina de almacenamiento o la herramienta de *backup*, que usan credenciales genéricas, se guardan todas en el gestor de claves **SysPass**[69]. Como se puede observar en la figura 6.1, existen dos entradas creadas para poder acceder al **vCenter** y a la herramienta de *backup* de **Veeam**.

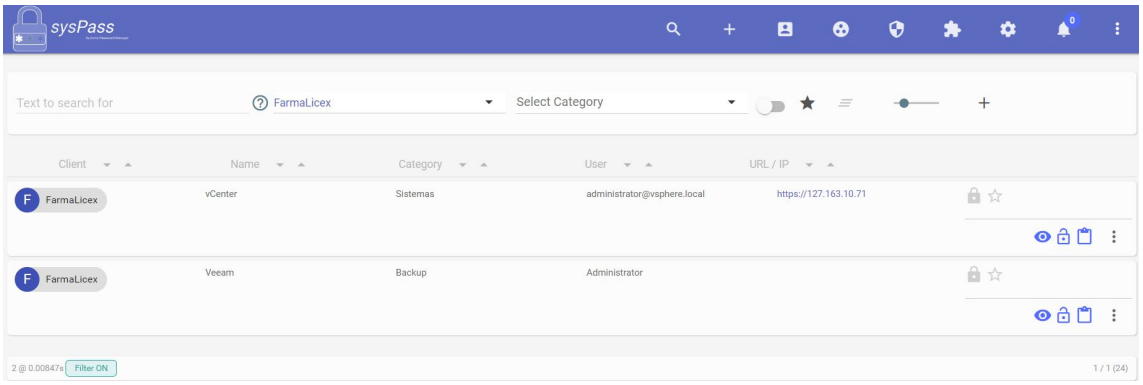


Figura 6.1: Ejemplo de dos cuentas almacenadas en **Syspass**

Para poder acceder a los servidores, el acceso se va a realizar por escritorio remoto usando la herramienta **RDM**, donde se añadirán todos los servidores, pero, para poder acceder a los servidores, se va a necesitar usar una VPN para conectarse a la red interna de **FarmaLicex**. Esto se va a realizar usando la herramienta gratuita **FortiClient VPN**, donde se va a utilizar el usuario de dominio mencionado anteriormente. Cuando se intente realizar la conexión, llegará un código o *token* al correo que tendrá que ser utilizado en un corto lapso de tiempo para poder validar la conexión. Posteriormente, ya se podrá acceder a los servidores con las credenciales. Una posible configuración de la VPN se muestra en la figura 6.2.

El parcheo de servidores está planificado para realizarse mensualmente. Esta planificación se va a dividir en tres grupos:

1. **Grupo de choque:** en este grupo se probarán los parches acumulativos de Windows Update que se vayan a instalar, además de otros posibles parches o actualizaciones que sean necesarias. Este grupo servirá para probar la instalación de los parches y tener margen de maniobra por si hubiera algún problema antes de instalar los parches en el resto de servidores.

2. **Servidores críticos:** en este grupo se van a parchear los servidores más importantes para la organización para tenerlos actualizados y protegidos de una manera eficaz y rápida.

[69] Herramienta para la gestión de contraseñas de forma segura. Consultar en: https://www.syspass.org/en.

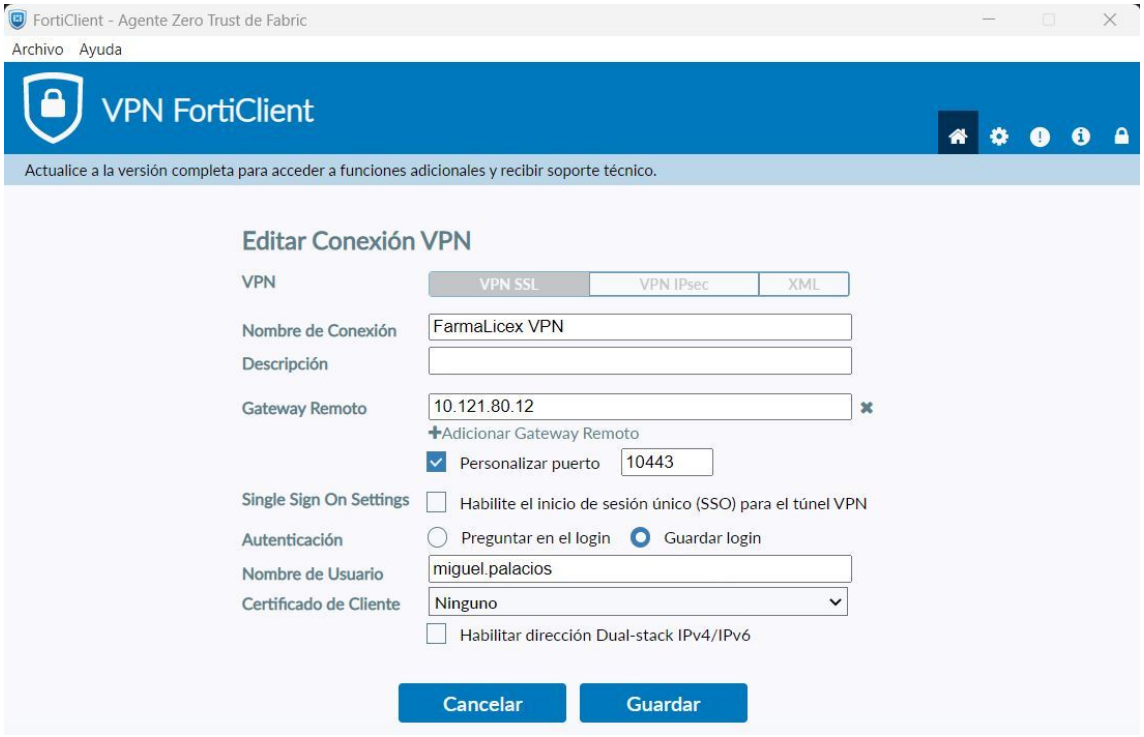


Figura 6.2: Configurar la VPN con **FortiClient**

3. **Resto de servidores:** en este grupo se van a parchear los servidores restantes una vez se hayan parcheado ya los servidores críticos.

Estos tres grupos tendrán su ventana de indisponibilidad durante el fin de semana por la mañana, para evitar cortes en la producción de **FarmaLicex**.

De esta forma, Alice y Bob, acuerdan que el ciclo mensual de parcheo de servidores se distribuya como se muestra en la tabla 6.1. El ciclo de parcheo comenzaría el segundo martes de cada mes, con el conocido *Patch Tuesday* donde se publicarían los parches. Al día siguiente, el equipo de ciberseguridad, revisará los parches por si hubiera surgido alguna vulnerabilidad *Zero-day*. El jueves, con la herramienta de ticketing **SD+**, el equipo de operaciones de **BobTec Soluciones** creará los cambios pertinentes para los parcheos del fin de semana del grupo de choque, y si no hubiera ningún problema el CAB aprobaría los cambios. Durante el fin de semana se realizará el parcheo en cuestión. Este ciclo se reparte en tres semanas, con los tres grupos creados anteriormente. Cada semana antes de crear los cambios pertinentes del fin de semana, se revisará en la página de **Microsoft** que no hayan descubierto ningún problema conocido de los parches o alguna vulnerabilidad que mitigar.

Esta planificación se enviará al cliente cada mes para que la valide y dé el visto bueno a proceder con el ciclo de parcheo mensual. La planificación a enviar será como la que se muestra en la Figura 6.3. En este documento, a modo de registro interno, se creará una hoja donde se apuntarán todos los parches instalados en cada servidor, con la fecha de instalación, ID del parche, descripción y una captura de pantalla donde se vea el parche instalado ya sea desde **Windows Powershell**, panel de control o Windows Update.

En el documento mostrado en la figura 6.3 se muestra como están agrupados los servidores, su estado de la última actualización, su dirección IP, el servicio o función que desempeñan dentro de la organización, su ventana de reinicio, su SO y, por último, una columna con observaciones por si las hubiera.

Tabla 6.1: Planificación del parcheo de servidores para **FarmaLicex**

Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo	Nº Semana
							Semana 1
	Publicación KB	Revisión KB	Aprobación GC		Instalación de las actualizaciones en GC		Semana 2
			Revisión y aprobación servidores críticos		Instalación de las actualizaciones en los servidores críticos		Semana 3
			Revisión y aprobación de resto de servidores		Instalación de las actualizaciones en el resto de servidores		Semana 4

Para poder monitorizar todos los activos de **FarmaLicex**, se instalará el agente de **Zabbix** en todos sus activos, para así poder estar al tanto de alertas, problemas, o utilizar la herramienta para visualizar gráficas de rendimiento y ver posibles tendencias de cara al futuro. De esta forma el equipo de monitorización o equipo *24x7* podrá monitorizar correctamente.

Nombre Servidor	Próxima Actualización	Nº Semana	Última actualización	IP	Servicio/función	Ventana de reinicio	Sistema Operativo	Observaciones
SRV-TEST01	16/6/2024	Semana 1 Dia 1	OK	192.168.144.149		Domingo de 10:45h a 13:45h	Windows Server 2019 Std	
SRV-TEST02	16/6/2024	Semana 1 Dia 1	OK	192.168.144.150		Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-MISTDC01	22/6/2024	Semana 2 Dia 1	OK	192.168.144.144	DC	Sábado de 10:45h a 13:45h	Windows Server 2019 Standard	
SRV-FUTVEEAM01	22/6/2024	Semana 2 Dia 1	OK	192.168.144.145	Servidor de *backup*	Sábado de 10:45h a 13:45h	Windows Server 2022 Std 21H2	Revisar *backups*
SRV-BBDD	22/6/2024	Semana 2 Dia 1	OK	192.168.144.146	BBDD Oracle	Sábado de 10:45h a 13:45h	Windows Server 2019 Std	
SRV-APP01	22/6/2024	Semana 2 Dia 1	OK	192.168.144.147	Servidor de aplicaciones	Sábado de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-MISTFTP	22/6/2024	Semana 2 Dia 1	OK	192.168.144.148	FTP	Sábado de 10:45h a 13:45h	Windows Server 2019 Standard	
SRV-MISTDC02	23/6/2024	Semana 2 Dia 2	OK	192.168.70.28	DC	Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	Físico
SRV-SQL14	23/6/2024	Semana 2 Dia 2	OK	192.168.70.29	BBDD SQL	Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	Parar servicio MySQL
SRV-NAV	23/6/2024	Semana 2 Dia 2	OK	192.168.70.30	Navision	Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-MISTFS01	23/6/2024	Semana 2 Dia 2	OK	192.168.70.31	Servidor de ficheros	Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-APP02	23/6/2024	Semana 2 Dia 2	OK	192.168.70.32	Servidor de aplicaciones	Domingo de 10:45h a 13:45h	Windows Server 2019 Standard	
SRV-TOOL	26/11/2023	Semana 2 Dia 2	OK	192.168.70.33	Farmatools	Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-LAB01	29/6/2024	Semana 3 Dia 1	OK	192.168.45.2		Sábado de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-LAB02	29/6/2024	Semana 3 Dia 1	OK	192.168.45.3		Sábado de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-DESARROLLO01	29/6/2024	Semana 3 Dia 1	OK	192.168.45.4		Sábado de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-PRINT	29/6/2024	Semana 3 Dia 1	OK	192.168.45.5		Sábado de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-LAB03	30/6/2024	Semana 3 Dia 2	OK	192.168.45.6		Domingo de 10:45h a 13:45h	Windows Server 2019 Std	
SRV-LAB04	30/6/2024	Semana 3 Dia 2	OK	192.168.45.7		Domingo de 10:45h a 13:45h	Windows Server 2019 Std	
SRV-DESARROLLO02	30/6/2024	Semana 3 Dia 2	OK	192.168.45.8		Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	
SRV-CHAIN	30/6/2024	Semana 3 Dia 2	OK	192.168.45.9		Domingo de 10:45h a 13:45h	Windows Server 2022 Std 21H2	Físico

Figura 6.3: Planificación para el parcheo de servidores de **FarmaLicex**

El parcheo de servidores se va a automatizar con la herramienta **ManageEngine Endpoint Central**, herramienta desde la cual se van a crear cinco grupos personalizados para la división mostrada en la imagen 6.3. En la figura 6.4, se muestra el grupo referente a la semana uno siendo el grupo de choque, y se muestran también para las semanas dos y tres dividido en dos días, siendo estos, sábado y domingo. Asimismo, se puede observar el número de miembros que tiene cada grupo, coincidiendo estos con los servidores planificados para cada día.

Después de haber creado los grupos, se crean las directivas de implementación, fijando la hora de parcheo desde las once menos cuarto de la mañana, hasta las dos menos cuarto de la tarde, dando margen suficiente por si hubiera que realizar algún plan de marcha atrás o intentar solucionar algún problema.

Por último, faltaría crear las implementaciones automáticas, aplicadas a los grupos y directivas creadas. En este caso se van a seleccionar solo los parches acumulativos de Windows Update para los Windows Server 2019 estándar y Windows Server 2022 21H2. Si se quisiera añadir algún parche más por petición de **FarmaLicex**, se podrían modificar estas implementaciones y añadirlos, o crear una nueva específica para esos parches. Las implementaciones automáticas ya creadas se muestran en 6.5.

Nombre	Tipo de recursos	Categoría de grupo	Miembros
semana			
Semana 1 Dia 1 (CHOQUE)	Equipos	Estático	2
Semana 2 Dia 1	Equipos	Estático	5
Semana 2 Dia 2	Equipos	Estático	6
Semana 3 Dia 1	Equipos	Estático	4
Semana 3 Dia 2	Equipos	Estático	4

Figura 6.4: Creación grupos para el parcheo de servidores de **FarmaLicex** desde **Endpoint Central**

Nombre	Hora de implantación	Hora de creación	Estado actual	Acción	Nº total de destinos
01 Semana 1 Dia 1	10:45 to 13:45	26/09/2023 11:00:58	1 1		2
02 Semana 2 Dia 1	10:45 to 13:45	19/10/2023 16:45:32	5		5
03 Semana 2 Dia 2	10:45 to 13:45	19/10/2023 16:54:36	6		6
04 Semana 3 Dia 1	10:45 to 13:45	26/09/2023 11:09:55	4		4
05 Semana 3 Dia 2	10:45 to 13:45	26/09/2023 11:14:28	4		4

Figura 6.5: Implementación automática para el parcheo de servidores de **FarmaLicex** desde **Endpoint Central**

Como se ha mostrado en la figura 6.3, para ciertos servidores hay algunas observaciones a tener en cuenta, como por ejemplo para los dos servidores físicos, esa misma mañana hay que comprobar que el *backup* se haya completado correctamente para tener un plan de marcha atrás. De igual manera, el servidor *SRV-MISTDC02*, además de ser un servidor físico es un DC, por lo que si fallara habría que aplicar el plan de marcha atrás explicado en el apartado 3.5.3. Para el servidor *SRV-SQL14* se debe parar el servicio manualmente antes de reiniciar el servidor y comprobar que posterior al parcheo el servicio esté activo. Por último para el servidor *SRV-FUTVEEAM01*, se debe revisar que no haya *backups* en marcha en su ventana de reinicio, y en caso de que haya, se debe parar el *backup*, para volver a activarlo después del parcheo.

Para el control de todas las máquinas virtuales de **FarmaLicex**, se va a usar **VMware vCenter**, herramienta desde la que se van a poder gestionar todos los servidores y recursos de forma centralizada. También es la herramienta desde la que se van a realizar las *snapshots* previas al parcheo.

Tras toda la planificación inicial se procederá a realizar el parcheo de forma mensual. Es clave tener un contacto estrecho con el cliente, informándole mensualmente de si se varía la planificación, de los nuevos parches que se van a aplicar o si tienen algún problema conocido. Además, se debe contactar con el cliente previa y posteriormente al parcheo de cada fin de semana.

CAPÍTULO 7
Conclusiones

Para finalizar con este TFG, se van a detallar los objetivos y problemas encontrados, la relación del trabajo con los estudios cursados y los posibles trabajos a futuro. Estos tres apartados son fundamentales a modo de resumen para entender el TFG.

7.1 Cumplimiento de objetivos y problemas encontrados

El objetivo de este trabajo ha sido poder diseñar una metodología integral para el parcheo de servidores en la que se abarcaran todos los aspectos importantes, mostrando las pautas a seguir y distintos apartados a tratar para lograr una planificación correcta. Además, era de suma importancia poder mostrar la diversidad tecnológica actual, exponiendo diferentes formas de agrupación de servidores y casos especiales a tener en cuenta. Otro objetivo importante de este trabajo era el de explorar distintas herramientas de automatización, siendo este un aspecto clave para lograr optimizar y ahorrar recursos a la hora de realizar el parcheo. Con todos estos aspectos ya cohesionados, se buscaría realizar un ejemplo realista para ver el proceso de una forma natural, logran así contribuir al conocimiento de la seguridad informática. Estos objetivos que acabo de mencionar, se han completado correctamente gracias a cumplir una metodología detallada en este proceso. He de remarcar la importancia de los dos diagramas que se han realizado en este proyecto, para los capítulos de planificación y ejecución del parcheo. El diagrama 3.1 sirve de gran apoyo visual a la hora de comenzar el capítulo de la planificación, puesto que se abarcan todos los temas que se van a tratar y las posibles relaciones que hay entre los distintos apartados, sirviendo así de guía para el lector, además de ofrecer un contexto amplio del tema. El diagrama 4.1, representa el diagrama de flujo a seguir a la hora de la ejecución del parcheo de servidores, donde se indican todos los caminos posibles, decisiones y acciones a realizar. Es un diagrama excelente como guía a la hora de ejecutar correctamente el parcheo de servidores, además de un gran apoyo visual para relacionarlo con todos los conceptos del capítulo.

Han sido varios los problemas que han surgido a lo largo del desarrollo del trabajo, pero que con paciencia, esfuerzo y una planificación correcta ha sido posible superarlos. El primer problema que me encontré a la hora de realizar este trabajo, fue no saber bien qué información usar y cómo plasmarla en el trabajo para que fuera una metodología general y no un caso específico. Gracias a la lectura de varios libros y artículos conseguí ese enfoque. Otro problema fue el hecho de explicar la sección del plan de emergencia (sección 3.5), debido a la complejidad de intentar condensar toda la información del plan de marcha atrás de los controladores de dominio o nodos físicos de un hipervisor en una sección. Además, el hecho de no poder ejemplificar estos casos, debido a la baja probabi-

lidad de que ocurran estos desastres, sumado a la criticidad de tener que solucionar esos problemas, imposibilita el hecho de mostrarlo.

7.2 Relación del trabajo desarrollado con los estudios cursados

Durante la confección de este TFG, se han aplicado y coordinado los conocimientos aprendidos durante la carrera. A continuación, se van a especificar las asignaturas y competencias transversales (CT) más relevantes para este proyecto:

- **Seguridad en los sistemas informáticos:** esta asignatura está intrínsecamente relacionada con este trabajo, puesto que su objetivo es contribuir al conocimiento en la seguridad informática, además de mejorarla. Con esta asignatura se aprende el concepto de la ciberseguridad, qué posibles ataques hay, cómo ser resiliente a fallos y cómo recuperarse de ellos. Estos aspectos se abarcan y se intentan explicar en este TFG.
- **Gestión de proyectos:** esta asignatura también ha sido importante a la hora de confeccionar el trabajo, al ser una asignatura directamente orientada al sector empresarial. Esta asignatura ha servido de base para saber establecer el proyecto. También ha sido de gran utilidad en la definición del alcance conjunto al cliente, o en cómo tratar con todas las partes interesadas, teniendo en cuenta riesgos y prioridades.

En cuanto a competencias transversales, todas han sido de gran utilidad tanto en mi desarrollo personal, como a la hora de desarrollar este trabajo, pero destacaría las siguientes:

- **CT-5 Diseño y proyecto:** esta CT ha servido como base a la hora de diseñar este trabajo e intentar cuadrar todos los apartados del proyecto.
- **CT-8 Comunicación efectiva:** en este TFG es importante esta CT, ya que esta metodología debe ser clara y concisa para que sea efectiva, además de enfatizar la importancia de la comunicación con el cliente.
- **CT-12 Planificación y gestión del tiempo:** esta CT ha sido de gran utilidad tanto a la hora de planificar mi TFG marcando los tiempos para cada apartado, como para cuadrar la planificación con el cliente.

7.3 Trabajos futuros

Aunque se hayan alcanzado los objetivos principales en este TFG, existen varias vías de mejora y ampliación de esta metodología para poder realizarse en un futuro. Estas mejoras son las siguientes:

- **Flecos pendientes y limitaciones de tiempo:** aunque el trabajo se haya completado correctamente, hay varios frentes que me hubiera gustado incluir en este TFG, pero que no ha sido posible debido a limitaciones de tiempo y a limitaciones de recursos, por ejemplo, me hubiera gustado poder ejemplificar el plan de marcha atrás de todas las opciones mostradas o haber mostrado el uso de más herramientas de automatización.

- **Mejoras posibles:** una mejora viable para este trabajo sería incluir en la metodología otros sistemas operativos que sean importantes en el ámbito empresarial, como por ejemplo, Linux. De esta forma, se explicaría cómo realizar la aplicación de parches en ese SO, sus ventajas y desventajas comparado con Windows, además de mostrar sus planes de marcha atrás.
- **Caminos a evitar:** es de gran importancia remarcar posibles estrategias de desarrollo que no serían beneficiosas para esta metodología. La principal sería extender esta metodología a otros dispositivos como móviles o electrónica de red como *switches*. Esto es debido a que el enfoque sería muy distinto y no habría sinergia entre las distintas partes, resultando en dos metodologías sueltas en lugar de una compacta.

Bibliografía

[1] BRYNJOLFSSON, Erik & MCAFEE, Andrew. *The second machine age: work, progress, and prosperity in a time of brilliant technologies.* Norton & Company Limited, W. W., 2016. ISBN 9780393350647.

[2] AGUILERA LÓPEZ, Purificación. *Seguridad informática.* Editex, 2010. ISBN 8497716574.

[3] Arango, D.M.E. y CANO, Y.L.J, 2022. Informe definitivo evaluación independiente procedimiento gestión de los servicios de TI. [en línea]. [consultado el 22 de junio de 2024]. Disponible en: https://www.cdm.gov.co/cgm/Paginaweb/IP/Reportes%20de%20control%20interno%202022/Informe%20Definitivo%20Evaluaci%C3%B3n%20Independiente%20Seguridad%20DT.pdf

[4] HERNÁNDEZ SAUCEDO, Ana Laura y MEJIA MIRANDA, Jezreel. Guía de ataques, vulnerabilidades, técnicas y herramientas para aplicaciones web. *ReCIBE.* [en línea] 2015, Año 4(1) [consultado el 22 de junio de 2024]. Disponible en: https://www.redalyc.org/pdf/5122/512251501005.pdf

[5] TORRES, Begoña, 2020. Sothis, la tecnológica valenciana que cautivó a Juan Roig. *Valencia Plaza* [en línea]. 25 de junio. Disponible en: https://valenciaplaza.com/tecnologica-sothis-cutiva-juan-roig [consultado el 22 de junio de 2024].

[6] EL MUNDO, 2022. Juan Roig vende el 100 % de Sothis a Nunsys para generar una gran tecnológica referente nacional. *El Mundo* [en línea]. 18 de febrero. Disponible en: https://www.elmundo.es/comunidad-valenciana/2022/02/18/620fbb8721efa09d288b45a1.html [consultado el 22 de junio de 2024].

[7] VOLDAN, Daniel, 2003 A Practical Methodology for Implementing a Patch management Process. *SANS Institute SANS Security Insights* [en línea]. [consultado el 22 de junio de 2024]. Disponible en: https://www.sans.org/white-papers/1206/

[8] SOUPPAYA, Murugiah. & SCARFONE, Karen, 2022. Guide to enterprise patch management planning: Preventive maintenance for technology. *National Institute of Standards and Technology. Special Publication* [en línea].[consultado el 22 de junio de 2024]. Disponible en: https://doi.org/10.6028/NIST.SP.800-40r4

[9] DIAMOND, T., KERMAN, A., SOUPPAYA, M., STINE, K., JOHNSON, B., PELOQUIN, C., RUFFIN, V., SIMOS, M., SWEENEY, S. & SCARFONE, K, 2022. Improving enterprise patching for general IT systems: Utilizing existing tools and performing processes in better ways. *National Institute of Standards and Technology. Special Publication* [en línea]. [consultado el 22 de junio de 2024]. Disponible en: https://doi.org/10.6028/NIST.SP.1800-31

[10] HOEHL, Michael, 2018. Agile Security Patching. *SANS Institute SANS Security Insights* [en línea]. [consultado el 22 de junio de 2024]. Disponible en: https://www.sans.org/white-papers/38410/

[11] MATEU SANCHEZ, Francisco Manuel, 2014. *Definición de una metodología ligera para la evaluación, implantación y gestión de la seguridad en sistemas informáticos* [en línea]. Trabajo fin de máster. Valencia: Universidad Politécnica de Valencia [consultado el 22 de junio de 2024]. Disponible en: http://hdl.handle.net/10251/52288

[12] KOUTRAS, V.P. & PLATIS, A.N., 2020. Software rejuvenation: Key concepts and granularity. *Handbook of Software Aging and Rejuvenation*. S.l.: WORLD SCIENTIFIC, pp. 41–70. ISBN 9789811214561.

[13] Actualizaciones de pila de mantenimiento. *Microsoft Learn* [en línea], [sin fecha]. [consultado el 22 de junio de 2024]. Disponible en: https://learn.microsoft.com/es-es/windows/deployment/update/servicing-stack-updates

[14] ¿Qué es un hipervisor? Redhat.com [en línea], [sin fecha]. [consultado el 22 de junio de 2024]. Disponible en: https://www.redhat.com/es/topics/virtualization/what-is-a-hypervisor

[15] Hipervisores: definición, tipos y soluciones. *Stackscale* [en línea], 2024. [consultado el 22 de junio 2024]. Disponible en: https://www.stackscale.com/es/blog/hipervisores/

[16] The 2020 state of virtualization technology. *SpiceWorks* [en línea], 2019. [consultado el 22 de junio 2024]. Disponible en: https://www.spiceworks.com/sw-marketing/reports/state-of-virtualization/

[17] Controlador de dominio. *Wikipedia* [en línea], [sin fecha]. [consultado el 22 de junio 2024]. Disponible en: https://es.wikipedia.org/wiki/Controlador_de_dominio

[18] ¿Qué es y para qué sirve un controlador de dominio? *TutoManiac* [en línea], 2023. [consultado el 22 junio 2024]. Disponible en: https://tutomaniac.com/que-es-y-para-que-sirve-un-controlador-de-dominio/

[19] Uso de GPOs (Group Policy Object) en Windows Server. *Solvetic* [en línea], 2016. [consultado el 22 junio 2024]. Disponible en: https://www.solvetic.com/tutoriales/article/2416-uso-de-gpos-group-policy-object-en-windows-server/

[20] MEHRTENS, Matthias, 2023 Mejores prácticas de backup para servidores físicos y Windows. *Veeam Software* [en línea]. [consultado el 22 junio 2024]. Disponible en: https://www.veeam.com/es/resources/wp-windows-physical-servers.html

[21] BEZET-TORRES, Jérôme. y BONNET, Nicolas, 2017. *Windows Server 2016: infraestructura de red: preparación para la certificación MCSA: examen n° 70-741: 23 trabajos prácticos, 86 preguntas-respuestas*. S.l.: Ediciones ENI. ISBN 9782409010798.

[22] ¿Por qué no debo utilizar una copia instantánea en lugar de un backup? *TECH2BUSINESS* [en línea], 2021. [consultado el 22 junio 2024]. Disponible en: https://t2b.tech/por-que-no-debo-utilizar-copias-instantaneas-snapshots-en-lugar-de-backup/

[23] Best practices for using VMware snapshots in the vSphere environment. *VMware Knowledge Base* [en línea], [sin fecha]. [consultado el 22 junio 2024]. Disponible en: https://knowledge.broadcom.com/external/article?legacyId=1025279

[24] Restaurar un controlador de dominio virtualizado. *Microsoft Learn* [en línea], [sin fecha]. [consultado el 22 junio 2024]. Disponible en: https://learn.microsoft.com/es-es/windows-server/identity/ad-ds/manage/virtual-dc/restore-virtualized-domain-controller

[25] Virtualización de controladores de dominio con Hyper-V. *Microsoft Learn* [en línea], [sin fecha]. [consultado el 22 junio 2024]. Disponible en: https://learn.microsoft.com/es-es/windows-server/identity/ad-ds/get-started/virtual-dc/virtualized-domain-controllers-hyper-v

[26] Configuración de una instalación Server Core de Windows Server y Azure Stack HCI con la herramienta de configuración del servidor (SConfig). *Microsoft Learn* [en línea], [sin fecha]. [consultado el 22 junio 2024]. Disponible en: https://learn.microsoft.com/es-es/windows-server/administration/server-core/server-core-sconfig

[27] Windows Server Update Services (WSUS). *Microsoft Learn* [en línea], [sin fecha]. [consultado el 22 junio 2024]. Disponible en: https://learn.microsoft.com/es-es/windows-server/administration/windows-server-update-services/get-started/windows-server-update-services-wsus

APÉNDICE A
Objetivos de Desarrollo Sostenible

Reflexionando sobre la relación de los Objetivos de Desarrollo Sostenible de las Naciones Unidas con el Trabajo Final de Grado, se puede comprobar que la mayoría de los ODS no tienen ninguna relación con este TFG, sin embargo, hay que destacar una gran relación con dos de ellos, además de una menor relación con otros tres ODS, pudiéndose observar en la tabla A.1. Estos ODS en los que puedo realizar una aportación son los siguientes:

Tabla A.1: Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS)

Objetivos de Desarrollo Sostenible	**Alto**	**Medio**	**Bajo**	**No procede**
ODS 1. **Fin de la pobreza.**				X
ODS 2. **Hambre cero.**				X
ODS 3. **Salud y bienestar.**				X
ODS 4. **Educación de calidad.**				X
ODS 5. **Igualdad de género.**				X
ODS 6. **Agua limpia y saneamiento.**				X
ODS 7. **Energía asequible y no contaminante.**				X
ODS 8. **Trabajo decente y crecimiento económico.**	X			
ODS 9. **Industria, innovación e infraestructuras.**	X			
ODS 10. **Reducción de las desigualdades.**				X
ODS 11. **Ciudades y comunidades sostenibles.**				X
ODS 12. **Producción y consumo responsables.**			X	
ODS 13. **Acción por el clima.**			X	
ODS 14. **Vida submarina.**				X
ODS 15. **Vida de ecosistemas terrestres.**				X
ODS 16. **Paz, justicia e instituciones sólidas.**				X
ODS 17. **Alianzas para lograr objetivos.**			X	

- **ODS 8 - Trabajo decente y crecimiento económico:** este ODS tiene como finalidad aumentar el crecimiento económico, además de focalizarse en la productividad laboral. Con esta metodología para el parcheo de servidores, se facilita esto último, puesto que se consigue agilizar y automatizar ciertos procesos del parcheo de servidores, logrando un significante ahorro de recursos, consiguiendo de esta forma una mayor productividad y por ende un mayor crecimiento económico tanto para la consultora, como para el cliente que ha contratado los servicios.
- **ODS 9 - Industria, innovación e infraestructura:** este ODS profundiza en el concepto de la innovación y de la importancia de crear infraestructuras robustas. Este

TFG abarca el concepto de la innovación, para conseguir tener una metodología para el parcheo de servidores de una forma más esquemática, ágil y automatizada, en lugar de una forma más rudimentaria, realizándose siempre de forma manual, sin herramientas de apoyo y sin un plan tan estructurado. Además, el mantenimiento y actualización de los servidores es clave y puede ser crucial para que las infraestructuras tecnológicas funcionen de manera óptima.

En conclusión, esta metodología para el parcheo de servidores se puede vincular con varios Objetivos de Desarrollo Sostenible. Principalmente, este TFG contribuye al ODS 8, ya que con este trabajo lo que se pretende lograr es aumentar la efectividad de la empresa, reduciendo costes, ofreciendo un servicio de mayor calidad, logrando una mayor productividad y crecimiento económico de la empresa. Además, gracias a la planificación tan detallada, las optimizaciones que se proponen o las formas de automatizar el proceso, se consiguen infraestructuras flexibles, robustas y con gran innovación, así consiguiendo cumplir el ODS 9. También se puede relacionar con los ODS 12 y 13, ya que si se aumenta la eficacia operativa de los servidores, se puede lograr desperdiciar recursos digitales logrando reducir el consumo de energía de los servidores y de la empresa en general. Además, el 17 también tiene relación, ya que con este TFG se busca lograr alianzas entre la consulta ofreciendo sus servicios y el cliente que los contrata.

Gracias a incorporar estos ODS a la metodología del parcheo de servidores, se contribuye al plan de las Naciones Unidas y a la Agenda 2030, interrelacionando estos objetivos, logrando así una mayor eficiencia operativa en el día a día, además de buscar un futuro sostenible para todo el mundo.