ATMIRE

Title	Diseño de un sistema de criptografía asimétrica como mecanismo de seguridad en el envío de mensajes SMTP para la empresa de e-commerce del sector textil basado en NIST-FIPS 140-3
Item Type	http://purl.org/coar/resource_type/c_7a1f
Authors	Lu Hurtado, Samuel Martin;Villalon Vilchez, Gustavo Enrique
Publisher	Universidad Peruana de Ciencias Aplicadas (UPC)
Rights	https://purl.org/coar/access_right/c_abf2
Download date	2026-11-28 04:29:36
Item License	https://creativecommons.org/licenses/by-nc-sa/4.0/
Link to Item	https://hdl.handle.net/10757/690727

UNIVERSIDAD PERUANA DE CIENCIAS APLICADAS

FACULTAD DE INGENIERÍA

PROGRAMA ACADÉMICO DE INGENIERÍA DE REDES Y COMUNICACIONES

Diseño de un sistema de criptografía asimétrica como mecanismo de seguridad en el envío de mensajes SMTP para la empresa de e-commerce del sector textil basado en NIST-FIPS 140-3

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el título profesional de Ingeniero de Redes y Comunicaciones

AUTOR(ES)

Lu Hurtado, Samuel Martin	0009-0001-6600-1642
Villalon Vilchez, Gustavo Enrique	0009-0009-0976-1715

ASESOR(ES)

Seminario Garcia, Hernan Augusto	0000-0001-6971-355X

Lima, 09 de junio de 2026

DEDICATORIA

Le dedicamos el resultado de este trabajo a nuestras familias, las cuales siempre han estado brindándonos su apoyo incondicional en todo momento, gracias por enseñarnos a afrontar las dificultades de la mejor manera posible.

A nuestro asesor de tesis por siempre ofrecernos su apoyo en el desarrollo de este trabajo con sus palabras motivacionales.

AGRADECIMIENTOS

Agradecimientos a nuestra alma mater UPC que nos acogió como sus alumnos para formarnos de la mejor manera posible, a todos nuestros profesores con los que compartimos aula durante todos estos años, los agradecimientos para ellos por sus consejos, anécdotas y su apoyo para ser profesionales de éxito.

RESUMEN

Un criptosistema asimétrico es utilizado para fortalecer el envío de mensajes de email de manera segura, debido a que la información enviada viaja de manera segura hacia su destinatario, sin embargo, hay estándares que se utilizando para el cifrado de archivos como el RSA, el DES y AES, estos surgen debido a la necesidad de poder cifrar los datos, ya que el envío de información puede ser interceptado por los cibercriminales. Es aplicable por ejemplo para el envío de mensajes SMTP. El cual sería importante para fortalecer la confidencialidad e integridad de los emails enviados a través de este protocolo, especialmente en un entorno de comercio electrónico donde se manejan datos sensibles como información de servicios bancarios o datos sensibles de los clientes. Por otro lado, también se analizará la aplicación del estándar NIST-FIPS 140-3 es cual es la norma de seguridad establecida por la entidad gubernamental norteamericana NIST, el cual establece los lineamientos de seguridad criptográficos utilizados para datos sensibles. Esto incluye aspectos como la generación segura de claves, la protección de las claves privadas, la aplicación de algoritmos sólidos y la incorporación de mecanismos de autenticación y verificación. Para la empresa de e-commerce "DeusMarket" del sector textil, el diseño de un criptosistema asimétrico proporcionaría una capa extra de seguridad, en la transferencia de emails en los envíos de pedidos hacia sus clientes y proveedores. En resumen, este diseño sería una medida muy efectiva para mejorar la seguridad de la comunicación y proteger los datos sensibles.

Palabras clave: Criptografía. Encriptación, Algoritmo, Correo electrónico, MFT, SMTP.

ABSTRACT

An asymmetric cryptosystem is used to strengthen the sending of email messages in a secure way, because the information sent travels safely to its recipient, however, there are standards that are used for file encryption such as RSA, DES and AES, these arise due to the need to be able to encrypt the data, since the sending of information can be intercepted by cybercriminals. It is applicable, for example, for sending SMTP messages. Which would be important to strengthen the confidentiality and integrity of emails sent through this protocol, especially in an electronic commerce environment where sensitive data is handled, such as information from banking services or sensitive customer data. On the other hand, the application of the NIST-FIPS 140-3 standard will also be analyzed, which is the security standard established by the North American government entity NIST, which establishes the cryptographic security guidelines used for sensitive data. This includes aspects such as the secure generation of keys, the protection of private keys, the use of robust algorithms and the implementation of authentication and validation mechanisms. For the e-commerce company "DeusMarket" in the textile sector, the design of an asymmetric cryptosystem would provide an additional layer of security in the transfer of emails in order shipments to their customers and suppliers. In summary, this design would be a very effective measure to improve communication security and protect sensitive data.

Keywords: Cryptography, Encryption, Algorithm, Email, MFT, SMTP.

5TSP_u201821571_Lu_Villalon

INFORME DE ORIGINALIDAD

3%	2%	1%	1%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUD

FUENTES PRIMARIAS

1	repositorioacademico.upc.edu.pe	Fuente de Internet
2	Submitted to Escuela de Posgrado Newman	Trabajo del estudiante
3	Submitted to Pontificia Universidad Catolica del Ecuador - PUCE	Trabajo del estudiante
4	www.produceareequipa.gob.pe	Fuente de Internet
5	www.fesc.edu.co	Fuente de Internet
6	hdl.handle.net	Fuente de Internet
7	Submitted to Universidad Internacional de la Rioja	Trabajo del estudiante
8	www.clickbolivia.com	Fuente de Internet
9	Submitted to Universidad Estatal a Distancia	Trabajo del estudiante
10	dl.dropboxusercontent.com	Fuente de Internet
11	Leon Lomparte, Katia Regina. "Encriptacion RSA de archivos de texto.", Pontificia Universidad Catolica del Peru - CENTRUM Catolica (Peru), 2020	Publicación
12	Garcia, Alejandra Rivero. "Protocolos Para La Seguridad De La informacion En eHealth: Criptografia En Entornos mHeath", Universidad de La Laguna (Canary Islands, Spain), 2	Publicación
13	es.hmongwiki.com	Fuente de Internet
14	qdoc.tips	Fuente de Internet

TABLA DE CONTENIDOS

1. CAPITULO 1 14

1.1 Introducción 14
1.2 Organización Objetivo 14
1.3 Campo de Acción 15
1.4 Identificación del Problema 15
1.4.1 Situación Problemática 15
1.4.2 Problema Por Resolver 18
1.5 Objetivo General y Objetivos Específicos 18
1.5.1 Objetivo General 18
1.5.2 Objetivos Específicos 19
1.6 Mecanismos de Logro de los Objetivos 20
1.7 Justificación 21
1.8 Estado del Arte 21
1.8.1 Antecedes 21
1.8.2 Estándares en criptografía 24
1.8.2.1 Criptografía Simétrica 24
1.8.2.2 Algoritmos de la Criptografía Simétrica 25
1.8.2.3 Criptografía Asimétrica 26
1.8.3 Aplicación de Casos de Éxito 27
1.9 Aporte Académico 28

2. CAPITULO 2: MARCO TEORICO 30

2.1 Cifrado Asimétrico 30
2.2 HASH Funtion 30
2.3 GPG (GNU Privacy Guard) 31
2.4 Algoritmos de Cifrado Asimétrico 32
2.4.1 DES (Data Encryption Standard) 32
2.4.2 AES (Advanced Encryption Standard) 32
2.5 Algoritmo de Cifrado Asimétrico 34
2.5.1 RSA (Rivest-Shamir-Adleman) 34
2.6 Proceso de generación de firmas digitales 35
2.6.1 DSA (Algoritmo de firma digital) 36

2.7 Estándares de Cifrado 37
2.7.1 NIST FIPS-140-3 37
2.8 Protocolo de Correo Electrónico (Email) 38
2.8.1 SMTP (Simple Mail Transfer Protocol) 38
2.8.2 Los Procedimientos SMTP 39
2.8.3 Transacción de Correo SMTP 40
3. **CAPÍTULO 3: ANALISIS DEL PROBLEMA** 42
3.1 Definición de requerimientos 61
3.2 Interesados del proyecto 61
3.3 Clasificación de los Interesados 62
3.4 Requerimientos funcionales 63
4. **CAPÍTULO 4: DISEÑO DE LA SOLUCIÓN** 64
4.1 Analizar los riesgos asociados a los procesos de la empresa 64
4.1.1 Identificación de Activos de Información 67
4.1.1.1 Clasificación de la Información 67
4.1.1.2 Clasificación de acuerdo con la Confidencialidad 67
4.1.1.3 Clasificación de acuerdo con la Integridad 68
4.1.1.4 Clasificación de acuerdo con la Disponibilidad 68
4.1.2 Desarrollo del análisis de riesgos 72
4.1.3 Matriz de Riesgo 75
4.2 Diseñar el sistema de criptografía asimétrica 80
4.2.1 Especificación de Implementación de Encriptación 80
4.2.2 Definición de la Encriptación 80
4.2.3 Algoritmo de Cifrado Asimétrico 81
4.2.4 Hash Funtion 83
4.2.5 GNU Privacy Guard (GPG) 84
4.2.6 Algoritmos más utilizados de acuerdo con el estándar 84
4.2.7 Configuración de encriptación de las llaves utilizando GPG 85
4.2.8 Configuración del Asunto 85
4.2.9 Configuración del Algoritmos y Tamaño de claves 86
4.2.10 Modelo de Arquitectura del Criptosistema Asimétrico SMTP y MFT 87
4.2.11 Componentes de Arquitectura del Criptosistema 88

4.2.11.1 Servidor GPG 88
4.2.11.2 Cambio de Llaves 89
4.2.11.3 Firewall 89
4.3 Diseñar del sistema funcional para la codificación 90
4.3.1 Proceso de generación de clave para la criptografía de tipo asimétrica 90
4.3.1.1 Proceso de configuración de claves en el entorno GPG 90
4.3.1.2 Recuperación de la Información del Par de Claves Criptográficas 91
4.3.1.3 Configuración del entorno Python-GPG y firma digital de archivos 91
4.3.1.4 Firma de Archivos (nano firmados.py) 95
4.3.1.5 Proceso de Cifrado de Archivos mediante el Script encriptados.py 95
4.3.1.6 Proceso de Descifrado de Archivos de desencriptados.py 97
4.3.1.7 Verificar Archivos 100
4.3.1.8 Disponibilidad de los Scripts en todo el Sistema 102
4.3.1.9 Validación y Ejecución de Scripts 103
4.4 Diseño e implementación del sistema automatizado 109
4.4.1 Repositorios para la automatización en Python con AS2 109
4.4.2 Inicialización de un entorno virtual de trabajo 110
4.4.3 Configuración del Servidor Principal (DeusMarket) 110
4.4.3.1 Configuración de la cuenta administradora del Servidor 01 112
4.4.3.2 Generación de claves del Servidor 01 (DeusMarket) 113
4.4.3.3 Puesta en marcha del Servidor 01 (DeusMarket) 113
4.4.3.4 Configuración de la clave privada 114
4.4.3.5 Configuración de certificados públicos 117
4.4.3.6 Registro de la Organización (DeusMarket) 119
4.4.3.7 Registro del socio comercial de (DeusMarket) 121
4.4.4 Configuración del Servidor 02 (GamarraStore) 123
4.4.4.1 Creación de usuario administrador del Servidor 02 (GamarraStore) 123
4.4.4.2 Creación de llave pública y privada del Servidor 02 (GamarraStore) 124
4.4.4.3 Configuración de certificados públicos 126
4.4.4.4 Registro de la Organización (GamarraStore) 128
4.4.4.5 Configuración del socio comercial (GamarraStore) 130
5. **CAPÍTULO 5: VALIDACION DE LA SOLUCION** 132

5.1 Plan de Pruebas 132
5.2 Propósito de la Evaluación 133
5.3 Proceso de preparación de validación 134
5.3.1 Elaboración de prueba Objetivo E1 134
5.3.2 Elaboración de prueba Objetivo E2 142
5.3.3 Elaboración de prueba Objetivo E3 158
5.5.4 Elaboración de prueba Objetivo E4 162
6. **CONCLUSIONES Y RECOMENDACIONES** 167
6.1 Conclusión 167
6.2 Recomendaciones 168
7. **BIBLIOGRAFÍA** 169

ÍNDICE DE TABLAS

Tabla 1 Resultados del Proceso de Formación básica en seguridad informática. 42
Tabla 2 Tabla de Interesados del Proyecto 62
Tabla 3 Clasificación de Interesados 62
Tabla 4 Tabla de Requerimientos Funcionales 63
Tabla 5 Clasificación de los activos en función de su proceso y subproceso 69
Tabla 6 Clasificación de activos en función de su nivel de criticidad 70
Tabla 7 Clasificación de activos según su tipo 71
Tabla 8 Clasificación de activos en función de su categoría de uso 71
Tabla 9 Clasificación de activos de acuerdo con su nivel de impacto 72
Tabla 10 Tabla Comparativa de los Algoritmos de Cifrado AES, DES y RSA. 82
Tabla 11 Valores de Referencia 87
Tabla 12 Tabla de Objetivos Estratégicos 132
Tabla 13 Tabla de preparación de Objetivo Estratégico 1 134
Tabla 14 Política de Contraseña Segura 136
Tabla 15 Intento de acceso con contraseña débil 137
Tabla 16 Activación de MF 138
Tabla 17 Solicitud de Segundo Factor 139
Tabla 18 Acceso exitoso con MFA 140
Tabla 19 Resultados de los datos de acuerdo con los criterios establecidos 141
Tabla 20 Matriz riesgo de Probabilidad e Impacto 142
Tabla 21 Tabla Cuadro de preparación de Objetivo Estratégico2 143
Tabla 22 Resultados de la Validación de la arquitectura 156
Tabla 23 Tabla de niveles de Valores 156
Tabla 24 Tabla Cuadro de preparación de Objetivo Estratégico 3 158
Tabla 25 Tabla Cuadro de preparación de Objetivo Estratégico 4 162

ÍNDICE DE FIGURAS

Figura 1 Denuncias recibidas por delitos informáticos del Perú 16
Figura 2 Número de denuncias de ciberdelitos recibidas 17
Figura 3 Índice de riesgos globales para el 2022 18
Figura 4 Proceso de Cifrado Asimétrico 30
Figura 5 Proceso de Funcionamiento del Algoritmo Hash 31
Figura 6 Flujo de Encriptación 31
Figura 7 Comparación entre los métodos criptográficos DES & AES 33
Figura 8 Descripción del funcionamiento del algoritmo de cifrado AES 33
Figura 9 Implementación de cifrado híbrido con incorporación de firma digital 35
Figura 10 Proceso de funcionamiento del algoritmo de firma digital DSA 36
Figura 11 Modelo de Funcionamiento del Protocolo SMTP 39
Figura 12 Cuestionario sobre Ciberseguridad 43
Figura 13 Encuesta Global de Seguridad 44
Figura 14 Tabla descriptiva de los niveles de accesos a sitios web de riesgo elevado 45
Figura 15 Cabecera de mensaje de DeusMarket 46
Figura 16 Evidencia de exposición de passwords 47
Figura 17 Funcionamiento de MFT 48
Figura 18 Muestra de mensajes maliciosos o engaño 49
Figura 19 Flujo de ciberataque 50
Figura 20 Evidencia de archivos en formato plano 51
Figura 21 Evidencia de exposición de información sensible 51
Figura 22 Evidencia de Riesgos y Vulnerabilidades Identificadas. 52
Figura 23 Informe de evaluación del sitio web DeusMarket 54
Figura 24 Reporte de los Protocolos Implementados en el Sitio Web DeusMarket 55
Figura 25 Detalle de uso de protocoles en el site de DeusMarket 55
Figura 26 Esquema de ataque Poodle 56
Figura 27 Reporte del Uso de Paquetes de Cifrado. 57
Figura 28 Análisis del Paquete de Cifrado TLS 1.1. 58
Figura 29 Análisis del Paquete de Cifrado TLS 1.0. 58
Figura 30 Análisis del Paquete de Cifrado SSL 3.0. 58
Figura 31 Protocolos TLS luego de la solución para los ataques de tipo PODDLE 59
Figura 32 Protocolos TLS luego de la solución 59

Figura 33 Ciclo de Gestión para el Inventario de Activos de Información. 66
Figura 34 Tabla de evaluación del impacto cualitativo. 74
Figura 35 Matriz de evaluación cualitativa del impacto. 74
Figura 36 Arquitectura criptográfica asimétrica aplicada a SMTP y MFT 88
Figura 37 Evidencia del servicio de mensajería AS2 en funcionamiento. 114
Figura 38 Evidencia de la asignación de la clave privada en el sistema. 114
Figura 39 Selección del archivo PEM previamente generado 116
Figura 40 Evidencia de la selección de la clave configurada en el sistema. 116
Figura 41 Evidencia de la generación exitosa de la clave privada. 117
Figura 42 Evidencia de la configuración de los certificados públicos en el sistema. 117
Figura 43 Selección de la clave pública P2. 118
Figura 44 Creación de Organización DEUSMARKET 120
Figura 45 Evidencia del registro exitoso de la organización en el sistema. 121
Figura 46 Creación del socio comercial 122
Figura 47 Evidencia del registro del socio comercial (Partner) en el sistema. 123
Figura 48 Evidencia de creación de llave privada 125
Figura 49 Evidencia de la generación exitosa de la clave privada en el sistema. 126
Figura 50 Evidencia de asignación de certificados públicos 127
Figura 51 Evidencia de la configuración del certificado público. 128
Figura 52 Registro de la organización GamarraStore 129
Figura 53 Evidencia del registro exitoso de la organización dentro del sistema 130
Figura 54 Evidencia del registro del socio comercial (Partner) en el sistema 131
Figura 55 Evaluación Comparativa de Algoritmos 144
Figura 56 Pruebas de funcionamiento de algoritmos 145
Figura 57 Evaluación Comparativa de Algoritmos 146
Figura 58 Definición de los mecanismos y pruebas 146
Figura 59 Prueba de funcionamiento de Mecanismos 147
Figura 60 Proceso de gestión de claves y pruebas 148
Figura 61 Pruebas realizas en GPG 149
Figura 62 Evidencia Operativa de ciclo de vida de las claves 149
Figura 63 Prueba de Almacenamiento seguro 151
Figura 64 Verificar firma digital y prueba 152
Figura 65 Pruebas realizadas en cada etapa del flujo 153

Figura 66 Validación de la arquitectura criptográfica propuesta 154
Figura 67 Evidencia de Pruebas realizadas 155
Figura 68 Evidencia de envío y recepción de mensajes cifrados 163
Figura 69 Evidencia de selección de partner 164
Figura 70 Evidencia de selección para el envío de mensajes y archivos 165
Figura 71 Confirmación del envío del mensaje 165
Figura 72 Evidencia del envío del archivo a GamarraStore 166

1. CAPITULO 1

1.1 Introducción

Este estudio se enfoca en el desarrollo de un criptosistema asimétrico para mejorar la seguridad del envío y recepción de pedidos de DEUSMARKET mediante el protocolo MFT. Para ello usaremos el estándar GPG y algoritmos de cifrado asimétrico, el cual busca proporcionar seguridad de extremo a extremo para salvaguardar la información remitida por correo electrónico. Esto asegura que los correos electrónicos permanezcan encriptados, ya sea que estén almacenados de forma local o en la nube.

Debido a que el servicio de email es crucial para el intercambio de datos entre las empresas hoy en día, es esencial proteger los datos transmitidos por este medio. Esto es especialmente importante para grandes organizaciones y para aquellos que desean intercambiar información personal y confidencial. La información personal suele almacenarse en servidores internos o en nubes privadas o públicas, lo que plantea preocupaciones sobre la confidencialidad, integridad, disponibilidad y autorización.

Aunque existen varias opciones para proteger los datos, el cifrado es el método más utilizado. Sin embargo, el cifrado por sí solo no es suficiente para asegurar la seguridad de grandes cantidades de datos confidenciales. Así mismo, una vez que la información se traslada a la nube, el control directo sobre ellos se pierde. Por lo tanto, es necesario considerar blindar toda información que son almacenados en la nube.

En este trabajo se propone un método de cifrado de extremo a extremo ofreciendo seguridad total a la información del usuario. Los datos pueden cifrarse para garantizar la seguridad de sus archivos y documentos. Con este enfoque, se puede afirmar que el mecanismo ofrecido brinda una adecuada seguridad para acceder y preservar la integridad de la información almacenada a costo bajo. Además, se proporciona un mecanismo adecuado de validación de integridad firmado por el remitente.

1.2 Organización Objetivo

Es la empresa pymes de E-Commerce del sector textil DeusMarket ubicada en Lima Metropolitana. Esto se debe a que DeusMarket hace uso del servicio de transferencia de información de mensajes a través de la plataforma cloud de Google, el cual es utilizada por toda la organización. Así mismo, DeusMarket cuenta con más de 50 empleado en la actualidad.

El proyecto proporcionaría beneficios significativos a la empresa por una razón principal, ya que consiste en disminuir la posibilidad de una eventual divulgación de información.

Visión: Desarrollar nuestra tienda virtual ofreciendo una amplia selección de productos de una excelente calidad y un destacado servicio al consumidor de forma completa. Buscamos brindar una experiencia de compra práctica, rápida y confiable.

Misión: Lograr que nuestra plataforma virtual sea líder nacional en el E-commerce, satisfaciendo la necesidad de nuestros clientes.

Objetivos Estratégicos: Brindar acceso a cualquier producto que un cliente necesite de nuestro catalogo online y poder realizar la entrega en el tiempo estimado.

1.3 Campo de Acción

Nuestro campo de acción será la empresa de E-commerce del sector Textil DeusMarket en Lima Metropolitana.

Este proyecto de investigación tiene como finalidad desarrollar un criptosistema asimétrico con el enfoque de fortalecer los mecanismos de seguridad implementados en la plataforma de comercio electrónico DeusMarket.

1.4 Identificación del Problema

1.4.1 Situación Problemática

DeusMarket inició sus actividades en 2015 como una empresa de comercio de prendas de vestir y calzado a través de E-Commerce. A lo largo de sus siete años de historia, ha construido relaciones muy sólidas con sus clientes y se esfuerza constantemente por mejorar sus plataformas a fin de adaptarse a los cambios drásticos que trae este tipo de negocio de venta online. Esta es una organización con más de 50 empleados activos el cual mantiene comunicación permanente con sus clientes, así como el intercambio de información a través del protocolo SMTP para él envió del correo electrónico. Por lo tanto, se considera indispensable e importante para la comunicación de pedidos y contacto. Así mismo, también realiza transferencias información confidencial y privada hacia los bancos, como cuentas financieras y datos personales. Estos archivos son solo transferidos en texto plano utilizando MFT (Media File Transfer), siendo el problema principal y el requisito normativo de los bancos que los datos transmitidos por paquetes estén completamente cifrados de punto a punto.

Los problemas que surgen en DeusMarket por no aplicarse los mecanismos necesarios, son los siguientes:

- La privacidad es un punto importante relacionado con la transferencia de e-mails realizadas por Internet. Los mensajes son almacenados en los clientes de e-mail como: MS Outlook, Mozilla Thunderbird, antes de que pueda ser recibido por el destinatario. Si dicho mensaje no cuenta con encriptación, un ciberdelincuente, puede interceptar, leer o modificar el mensaje en cualquier punto de la transferencia durante el envío.

- Los servicios de email cloud no garantizan suficientemente la confidencialidad en los mensajes almacenados en su infraestructura, por lo que existe la necesidad de que estos deban de cifrarse de extremo a extremo con el fin de poder contar con un idóneo nivel de resguardo de la información confidencial, así como con los socios de la empresa.

- Incidentes de seguridad que impliquen pérdida o hurto de equipos o robo de información privada (como lo indica las estadísticas de denuncias recibidas por la PNP-2021), el cual conlleva graves consecuencias no solo financieras o legales, sino con multas por el incumplimiento de la legislación peruana N° 29733, pero también puede provocar pérdida de clientes causado por una reputación o mala imagen.

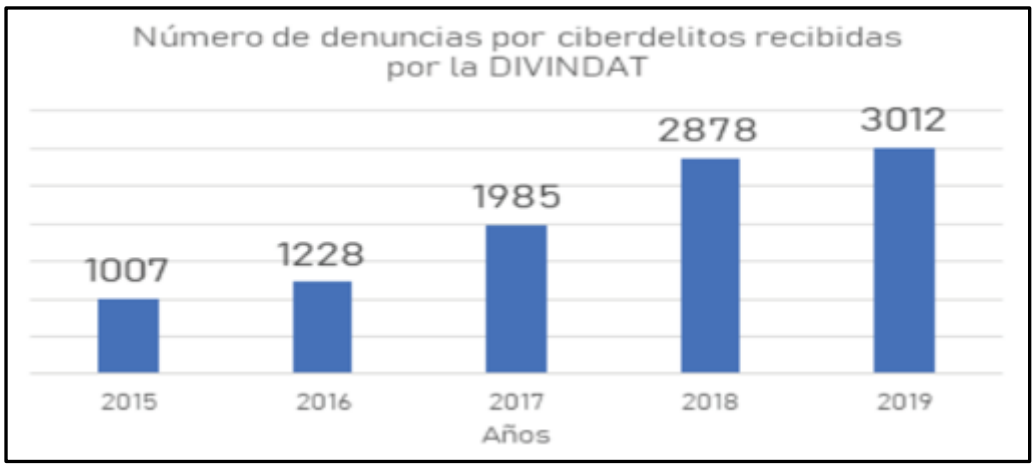
Figura 1

Denuncias recibidas por delitos informáticos del Perú



- El crecimiento exponencial de la ciberdelincuencia en los últimos años (Como lo indica la DIVINDAT en los últimos 5 años) y la facilidad para encontrar herramientas realizar ciberataques dirigidos, esto significa un mayor riesgo sobre cuentas de e-mail, especialmente para ejecutivos de empresas como DeusMarket, que cuenta con mucha información sensible de sus clientes y proveedores.

Figura 2
Número de denuncias de ciberdelitos recibidas

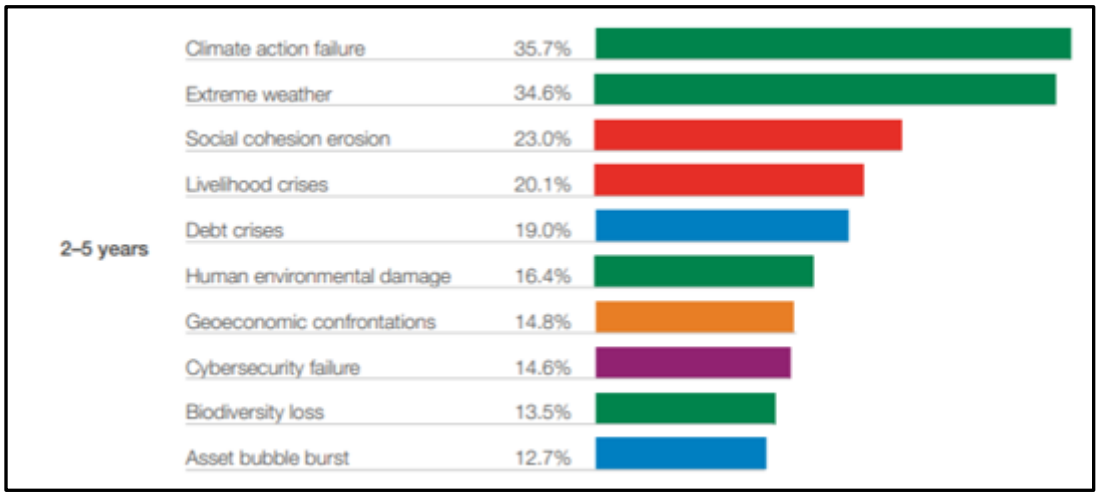


Nota: Desde el 2015 al 2019 se han incrementado los ciberdelitos. De "Número de denuncias por ciberdelitos recibidos por la DIVINDAT, por DIVINDAT, 2019 (Observatorio INDAGA)

- Los métodos tradicionales de identificación de contraseñas pueden verse comprometidos fácilmente con herramientas de fuerza bruta, por lo que los usuarios no tienen autenticación de doble factor (MFA) para verificar la identidad de su cuenta de correo electrónico.

Los indicadores de riesgos a nivel mundial para el 2022 del FEM, la falta de poder contar con una adecuada ciberseguridad está entre los 10 principales riesgos globales con más probabilidad de suceder y mayores consecuencias, esto se puede ver a diario con la gran cantidad de ataques cibernéticos que suceden.

Figura 3
Índice de riesgos globales para el 2022



Nota: De acuerdo con lo indicado por el foro mundial los niveles de ciberdelincuencia cuentan con un índice elevado. Por Foro Económico Mundial, 2022 (Foro Económico mundial)

Con este antecedente, la mayoría de las pymes en el Perú están dentro de las más vulnerables a crímenes cibernéticos, debido a que desconocen el impacto hacia su empresa, causando perdidas monetarias.

1.4.2 Problema Por Resolver

Deficiencia en los controles sobre la seguridad de almacenamiento y transición de información confidencial hacia y desde DeusMarket, así como el contacto con internos y externos. Esto puede provocar la exposición o fuga de información confidencial si los equipos de los usuarios sufren alguna perdida o hurto, además de los ataques realizados por delincuentes informáticos para robar información, estos intercambios de información confidencial y sensible es realizada por los usuarios, clientes y proveedores de la empresa.

1.5 Objetivo General y Objetivos Específicos

1.5.1 Objetivo General

Diseñar un sistema de criptografía asimétrica como mecanismo de seguridad para el envío de mensajes SMTP en una empresa de e-commerce del sector textil, de acuerdo con los lineamientos de la norma NIST FIPS 140-3, garantizando la protección de la información, veracidad de los datos y la accesibilidad de la información

1.5.2 Objetivos Específicos

OE1. Analizar los riesgos asociados al proceso de envío y almacenamiento de mensajes SMTP en una empresa de e-commerce del sector textil, identificando el 100% de los activos, amenazas y vulnerabilidades relevantes, en un plazo de 3 semanas.

OE2. Diseñar el sistema de criptografía asimétrica para el envío seguro de mensajes SMTP, especificando los algoritmos, la gestión de claves y los mecanismos de cifrado y firma digital, en un plazo de 4 semanas.

OE3. Diseñar el modelo funcional del sistema, incluyendo los procesos de codificación, envío y almacenamiento de mensajes, mediante diagramas y especificaciones técnicas, en un plazo de 4 semanas.

OE4. Diseñar e implementar un sistema automatizado orientado a la codificación por bloques y la transferencia de información mediante AS2, haciendo uso del lenguaje Python, definiendo su arquitectura lógica, flujo de procesamiento y mecanismos de comunicación segura, en un plazo de 4 semanas.

1.6 Mecanismos de Logro de los Objetivos

Objetivo Especifico	Indicador de Logro	Métrica
Analizar los riesgos asociados al proceso de envío y almacenamiento de mensajes SMTP en una empresa de e-commerce del sector textil, identificando el 100% de los activos, amenazas y vulnerabilidades relevantes, en un plazo de 3 semanas.	• Inventario de Activos • Matriz de Riesgos	• Porcentaje de numero de activos de riesgo. • Porcentaje de riesgos críticos
Diseñar el sistema de criptografía asimétrica para el envío seguro de mensajes SMTP, especificando los algoritmos, la gestión de claves y los mecanismos de cifrado y firma digital, en un plazo de 4 semanas.	• Arquitectura de proceso de operación de envió y almacenamiento de datos	• Disminución en porcentaje de horas de trabajo normal
Diseñar el modelo funcional del sistema, incluyendo los procesos de codificación, envío y almacenamiento de mensajes, mediante diagramas y especificaciones técnicas, en un plazo de 4 semanas.	• Programación de Script en Python	• Eficiencia del script en Python, porcentaje de aumento de archivos cifrados
Diseñar un sistema automatizado para la codificación por bloques y el envío de mensajes mediante el protocolo AS2, utilizando el lenguaje de programación Python, definiendo su arquitectura lógica, flujo de procesamiento y mecanismos de comunicación segura, en un plazo de 4 semanas.	• Sistemas de automatización en AS2 • Resultado de fases luego de la creación de mecanismos de encriptación	• Disminución en porcentaje de horas de trabajo manual

1.7 Justificación

Debido al aumento de ataques cibernéticos relacionados con el robo de información a través de mensajes electrónicos fraudulentos a nivel mundial, siendo básicamente al crecimiento del e-commerce como consecuencia de la pandemia que se viene presentado, el Perú no ha sido ajeno a ello y ha experimentado una serie de ciberataques graves (Diario Gestión, 2021) reportados en medios técnicos y publicados por TESSIAN (Tessian 2021) y KASPERSKY (SecureList LAT 2021), resulta imprescindible adquirir conocimiento acerca de los ataques más frecuentes dirigidos a los empleados de una empresa, en particular de DeusMarket. Esto permitirá tomar las medidas adecuadas para encriptar los correos electrónicos intercambiados, garantizando la privacidad y protección de toda la información.

Este trabajo de investigación busca el poder asegurar a protección y la integridad de las operaciones comerciales llevadas a cabo mediante SMTP y AS2, teniendo como objetivo garantizar la seguridad de su información. Así como también, el de proporcionar un diseño de un mecanismo o sistema criptográfico asimétrico que pueda aplicarse en la compañía para así blindar uno de los puntos débiles que se puede tener en una compañía, como es el usuario final, el cual normalmente es el más fácil de poder vulnerar al no conocer temas de ciberseguridad o estar al tanto de ello.

Dado que no existe una estrategia para informar o prevenir a los usuarios de este tipo de ataques, vale la pena señalar que este trabajo propone el utilizar el estándar GPG y algoritmos de encriptación basados en RSA simétricos. Así mismo, se busca el de poder ampliar las capas de seguridad implementadas por DeusMarket, el cual generará confianza a todos los usuarios de la organización a la hora de intercambiar datos sensibles vía SMTP y AS2.

Por último, presente proyecto es verdaderamente aplicable a DeusMarket. También, se han evaluado otras opciones, pero significan un alto costo en su implementación el cual no podría ser asumida por la compañía en este momento, pues es una empresa del sector PYME y no cuenta con un alto presupuesto para ello, siendo la mejor propuesta costo/beneficio el de poder aplicarse el uso de GPG.

1.8 Estado del Arte

1.8.1 Antecedes

A lo largo de la historia, la criptografía ha sido utilizada en diversas etapas, desde métodos antiguos como la escítala espartana hasta su aplicación por Julio César en la comunicación con sus generales. En la actualidad, la criptografía ha ampliado significativamente su campo de

aplicación, dejando de limitarse únicamente a los ámbitos militar y gubernamental para convertirse en una herramienta de uso cotidiano dentro de la sociedad. Su incorporación en diversos entornos tecnológicos ha permitido fortalecer la protección de la información y garantizar mayores niveles de seguridad en las comunicaciones digitales.

Dentro de este contexto, la criptografía de clave pública, denominada también criptografía asimétrica, se propone como una propuesta de solución ante la problemática de transmitir claves de forma segura por medios públicos. Este método se fundamenta en el empleo de dos claves relacionadas entre sí: una clave pública, la cual puede difundirse libremente, y una clave privada, cuyo acceso debe mantenerse exclusivamente restringido al propietario.

Asimismo, el crecimiento y desarrollo de Internet ha consolidado a la criptografía como un componente esencial en múltiples aplicaciones tecnológicas, entre ellas los certificados digitales y las firmas electrónicas. Gracias a estos mecanismos, es posible realizar transacciones electrónicas de manera segura y confiable.

En ese sentido, y considerando las necesidades planteadas en el presente proyecto, se evalúa la implementación de la criptografía GPG como mecanismo de seguridad informática. De acuerdo con las investigaciones revisadas, este tipo de tecnología contribuye significativamente al cumplimiento de los pilares de la seguridad informática y reduciendo la posibilidad de vulneraciones o accesos no autorizados.

Podemos indicar que, de acuerdo con lo revisado y con el enfoque en el proyecto, se puede evidenciar, en primer lugar, que el comercio electrónico a crecido años tras años en el mundo. Sin embargo, durante los últimos años, este sector ha aumentado enormemente puesto la prohibición de salir libremente a la calle, el periodo de confinamiento durante la emergencia sanitaria del año 2021 impulsó el uso del comercio electrónico, ya que las restricciones sanitarias fomentaron que las personas realizaran cada vez más compras a través de internet. (Mendoza Ybarra, C. A. (2021) indica que “es muy importante tener mecanismos correctos de seguridad informática sobre el e-commerce.” (p.27), como es el hecho de poder mejorar los procesos de pagos electrónicos (Mendoza Ybarra, C. A. (2021), así como el de poder identificar las vulnerabilidades que conlleva poder montar toda una plataforma de venta electrónica (Narbona, J. L. (2020), el cual pueda incluir un análisis más profundo a nivel de lo que pueda adolecer o contemplar para un adecuado proceso de todos los sistemas y procesos que puede considerarse (Olmedo, M. A. (2021).

En segundo lugar, en la categoría de Vulnerabilidades del Protocolo SMTP existen una investigación donde se realiza el análisis de Mecanismos de Defensa contra los ataques al protocolo SMTP Spoofing. Asimismo, otro de los estudios realizados nos indica que existen graves problemas de seguridad en los sistemas de correos electrónico a través del protocolo SMTP (Holst-Christensen, B., & Frokjar, E. (2021) nos menciona "el alcance que cuando un correo electrónico no viaja cifrado es vulnerable a ser interceptado por los atacantes." (p.55). Además (León Rodríguez, J. D. (2019) nos indica que "al viajar los mensajes cifrados mediante la encriptación dicha información viaja de manera segura. "(p.76). (De La Torre Yamberla, R. H. (2022) nos señala que "este es un estándar que exige que el correo que se envía al dominio esté autenticado y cifrado en muchos servidores SMTP." (p.64)

En tercer lugar, para la categoría de Aplicación de criptografía como mecanismos de Seguridad Informática, existen propuestas donde se propone realizar la encriptación del correo electrónico mediante las firmas digitales con el uso de la criptografía RSA (Gargallo Nieto, B. 2022), nos indica que el cifrado RSA es el primero y más utilizado algoritmo valido para firmas digitales. Asimismo, el uso de la aritmética modular como control de seguridad del sistema criptográfico RSA (Janampa Patilla, H. 2020), este tipo de criptografía hace uso de la matemática para la generación de claves publicas asimétricas RSA. (Colliard Schneider, C. D. (2019) nos indica que" también tenemos los protocolos criptográficos sustentados en los algoritmos de Diffie-Hellman & RSA." (p.24)

En cuarto lugar, debemos de indicar que siempre se tiene una constante búsqueda por encontrar o mejorar mecanismos de seguridad que puedan blindar nuestros sistemas o servicios informáticos, tal es así que se contempla el uso de la criptografía como se menciona en el segundo párrafo, pero ahora siendo más específicos, el poder analizar el uso de la criptografía basada en el protocolo de seguridad GPG, siento ese protocolo un mecanismo matemático (Plaza Martín, F. J. 2021), que busca el poder encriptar la comunicación entre dos extremos. También, es importante conocer que el uso o aplicación es de recurrir a llaves publicas GPG, justamente para poder realizar una transición de datos de forma cifrada (Ghasemi, P. 2015), blindando o asegurando todo dato durante la comunicación. Así mismo, se puede aplicar este control de seguridad sobre el plataformas o sistemas de comercio electrónico. Ahmad, K. & Alam, M. S. (2016) menciona que "ya que hoy por hoy, este sector como ya se ha hablado antes se encuentra en auge y es muy importante el poder implementar mecanismos que puedan hacer la venta electrónica mucho más segura y fiable, donde pueda brindar una total tranquilidad a los consumidores." (p.43)

En quinto lugar, también este mecanismo de encriptación se puede utilizar en los sistemas de correo electrónico. (Khan, W. U., & Arfeen, N. U. (2020) nos indica que "con el fin de fortalecer la seguridad en el envío de correos electrónicos, ya que ahora con el crecimiento de las telecomunicaciones, este es un medio muy usado día a día, tanto para el trabajo como por las escuelas, colegios, universidad en general." (p.37)

Esto propone, que cada vez exista mejores controles sobre la seguridad de los servicios de e-mail. (Qashqari, A., & Aljahdali, A. (2020) sostiene que "sin embargo, también es muy importante el de considerar un adecuado análisis y diseño que debe tener al momento de poder implementar mecanismos de cifrado sobre el correo electrónico. "(p.67)

Por último, en la categoría de NIST-FIPS Módulos de criptografía, tenemos los requisitos de seguridad, aquí podemos encontrar los requisitos para la implementación. (Cooper, M. J., & Schaffer, K. B. (2019) sostiene que "La validación del módulo completo FIPS 140-3 se refiere a la certificación que deben cumplir las aplicaciones que ejecutan funciones criptográficas especificadas." (p.45)

De igual manera tenemos el estándares Federal Information Processing Standards (FIPS) Boboň, S. (2021) india que "FIPS 140-3 donde se detalla el diseño y la ejecución de una herramienta automatizada diseñada para extraer información de manera eficiente de documentos relacionados con la seguridad FIPS 140-3 utilizando expresiones regulares, centrándose principalmente en la extracción de referencias de otros validados productos Luego se analiza la información extraída y se crea un gráfico dirigido de dependencias entre productos validados." (p.86)

1.8.2 Estándares en criptografía

1.8.2.1 Criptografía Simétrica

En el ámbito de la criptografía, el remitente y el destinatario emplean una única contraseña para la codificación y decodificación de la información. Al respecto, el autor Batten (2013) proporciona una explicación detallada.:

Los algoritmos de llave simétrica se basan en el proceso de descifrado el cual es simplemente la inversión del proceso de cifrado. Por ejemplo, si un mensaje se cifra desplazando cada letra un número estable de posiciones en el alfabeto, para descifrarlo se desplazan las letras en sentido contrario. En estos sistemas, tanto el remitente como el destinatario usan la misma llave. (p.3).

1.8.2.2 Algoritmos de la Criptografía Simétrica

AES

Hasan Jion (2019), define que:

AES (Avanzado Cifrado Standard) fue presentado por primera vez por Rijndael en octubre de 2000. Fue diseñado por Vincent Rijmen y Joan Daemen en Bélgica. Es un cifrado de bloque simétrico que puede tener diferentes tamaños de bloque, como 128, 192 y 256 bits. Esencialmente, los cálculos de cifrado se separan en tres clasificaciones notables; división simple por tonalidad, sustitución y transposición. AES utiliza una función de ronda que se basa en cuatro cambios distintos organizados por bytes, por ejemplo, Subbyte, Fila de desplazamiento, Columna de combinación, Agregar tecla de ronda.

DES

Hasan Jion (2019), afirma que:

DES (Estándar de cifrado de datos) fue desarrollado en 1970 en IBM por Horst Feistel. Este estándar de cifrado fue recomendado por el IEEE. DES se ocupa de la entrada de 64 bits de texto. El texto cifrado es de 64 bits. Independientemente de si la clave de 64 bits es la entrada genuina, la longitud de la clave es 56 para este cálculo. DES, realiza su operación por sustitución y permutación en total de 16 rondas. En consecuencia, los bits clave y de datos se permutan, se someten a XOR, se desplazan y luego pasan a través de 8 cajas. Seguir la tabla de búsqueda es un paso básico del algoritmo DES. El proceso de descifrado se realiza a la inversa. Esto lo convierte en un algoritmo de clave simétrica.

Blowfish

El algoritmo Blowfish utiliza un bloque de longitud variable con un tamaño de clave de 64 bits. Fue descubierto en 1993 por Bruce Schneier. En Blowfish, el tamaño de la llave puede variar de 32 a 448 bits. Este cálculo es una figura de Fiestel de 16 rondas y utiliza una enorme caja S de guarda clave. Blowfish requiere alrededor de 5 KB de memoria. Por regla general, Blowfish utiliza un tipo único de generación de claves. En la fase de expansión de claves, crea una tabla (4168 bytes) de subclaves convirtiendo la clave única hasta 448 bits. Esto genera un aumento de las subclaves.

GOST

Aktaş, H. (2018), nos dice:

Los algoritmos GOST (Gosudartsvennyi Standart) son algoritmos de seguridad estatal desarrollados por la Federación Rusa (antes Unión Soviética). El primero de estos algoritmos es el algoritmo de cifrado y descifrado GOST 28147-89 desarrollado en 1987. Otros algoritmos son el algoritmo de función hash GOST 34.11-94 y el algoritmo de firma digital GOST 34.10-2001. El algoritmo de cifrado GOST 2814789 es un algoritmo principal de encriptado de bloque de 64 bits y este algoritmo principal se utiliza en el algoritmo de función hash GOST 34.11-94 y en el algoritmo de firma digital GOST 34.10-2001. Dado que el tiempo computacional de los algoritmos de cifrado es muy alto, para hacer algoritmos de cifrado rápidos y en tiempo real, los FPGA (Field Programmable Gate Arrays) son las mejores plataformas para implementar estos algoritmos, excepto los ASIC (Application-Specific Integrated Circuits). En este estudio, el algoritmo de cifrado y descifrado GOST 28147-89 se implementará y se probará la velocidad del algoritmo para aplicaciones en tiempo real.

1.8.2.3 Criptografía Asimétrica

Dentro del campo de la criptografía asimétrica, se generan simultáneamente dos claves de cifrado: una contraseña pública que es de acceso libre y una contraseña privada que se utiliza para descifrar los mensajes encriptados con la clave pública.

DIFFIE-HELLMAN

Hasan Jion (2019), sostiene que: el protocolo Diffie-Hellman corresponde a un método de intercambio de claves criptográficas. Fue desarrollado por ambos autores (Whitfield Diffie y Martin Hellman) en 1976, proponiendo A y B definir una clave secreta incluso si E escucha su comunicación.

RSA

(Hasan J. (2019) señala que “RSA es un criptosistema asimétrico. Hay dos claves diversas. RSA es más conocido como criptosistema de clave pública, ya que uno de ellos se comparte con todos. La otra clave es privada ya que se mantiene en secreto. RSA se basa en hechos matemáticos.” (p. 78)

DSA (Digital Signature Algorithm)

Según Batten (2013), el DSA "este sistema se creó en los años 90 por el Instituto Nacional de Estándares y Tecnología (NIST) en colaboración con la Agencia de Seguridad Nacional de los Estados Unidos (NSA). Posteriormente, se sometió a revisiones en 1993, 1996 y nuevamente en el año 2000" (p. 119).

ECC

Hasan Jion (2019), La criptografía de curva elíptica fue presentada a mediados de la década de 1980 por Koblitz y Miller como una opción prometedora para las convenciones criptográficas a la luz del problema del logaritmo discreto en la recopilación multiplicativa de un campo limitado. ECC es lo mismo que RSA, pero diferente es que tiene una capacidad de resolución rápida y tiene una forma diferente de algoritmo criptográfico. La clave de seguridad de la ECC La longitud es más pequeña que otros algoritmos asimétricos y su tamaño de llave es de solo 163 bits. ECC y RSA toman tiempo exponencial completo y tiempo su exponencial respectivamente. Para un caso, si el tamaño de clave de RSA es m, entonces 1024 piezas requieren 4*10 millones de MIPS años con el asalto más conocido ECC con 160 piezas, el tamaño de clave requiere 12,6*10 millones de MIPS. Utiliza la ecuación de la curva electiva en lugar de los números primos tradicionales. La mayor parte del tiempo de ejecución se dedica a la multiplicación escalar. ECC proporciona la misma seguridad que otros algoritmos, pero en tamaños de clave más pequeños.

1.8.3 Aplicación de Casos de Éxito

SMARTMAIL

Matteo Duò (2022), Sostiene que:

SmartMail es una plataforma de correo electrónico segura que ayuda a las empresas a administrar cuentas y almacenar correos electrónicos en una caja fuerte encriptada. Los correos electrónicos se pueden cifrar automáticamente con OpenPGP y solo se pueden abrir con una contraseña. Permite a los administradores configurar dominios personalizados. Así mismo, esta solución/plataforma no permite almacenar ni acceder a las contraseñas de los usuarios privados y, si se olvidan las contraseñas, los empleados pueden solicitar un correo electrónico o un código de recuperación. Los usuarios, solo pueden ver una clave maestra una vez durante el proceso de registro y se puede usar para restablecer una contraseña de usuario, recuperar claves GPG y recuperar cuentas a través de la página de recuperación de cuentas. No es posible recuperar una clave maestra olvidada.

QUSECURE

Dave, K (2022) sostiene que:

El lanzamiento de QuSecure Nucleus, un revolucionario sistema de ciberseguridad post-cuántica. Nucleus, según la empresa, es la primera tecnología de extremo a extremo basada en software cuántico diseñada para salvaguardar comunicaciones y datos cifrados a través de un canal cuántico seguro. Según QuSecure, la solución hace frente a los ataques tradicionales existentes, así como a las futuras amenazas de la computación cuántica para organizaciones comerciales e instituciones gubernamentales.

FIPS 140-3

Wang Bin (2018) indica que

Otro logro significativo del programa de ciberseguridad de Hikvision ha sido el anuncio de la empresa de que ha obtenido la certificación FIPS 140-3 (Federal Government Information Processing Standard). El NIST, organismo perteneciente a la institución gubernamental responsable del comercio en los Estados Unidos, desarrolló el estándar de cifrado FIPS 140-3 del gobierno estadounidense. Además del sector comercial, industrias reguladas como la financiera, la sanitaria, la legal y la de servicios públicos también emplean FIPS 140-3, al igual que la CIA de Estados Unidos y Canadá. Esta empresa es la primera en obtener la certificación FIPS 140-3 para grabadores de vídeo en red (NVR) y cámaras IP.

PHOTON

Somdev, S. (2020) nos dice que:

Photon es una herramienta de open source escrito en Python para diseñar rastreadores web. Se trata de un rastreador de alto rendimiento desarrollado para herramientas de inteligencia de fuentes abiertas. El proyecto Photon facilita los procesos de recolección de información, permitiendo, además, la obtención de datos desde enlaces URL, direcciones de correo electrónico, perfiles en redes sociales, documentos en formato XML y PDF, así como también desde depósitos en Amazon S3.

1.9 Aporte Académico

En este documento, demostramos cómo se pueden implementar soluciones innovadoras y prácticas para proteger el almacenamiento de datos mediante el uso de algoritmos criptográficos. Para lograrlo, ofrecemos dos medidas de encriptación asimétrica para asegurar

los datos almacenados y transferidos a través de la red. Una de ellas es un algoritmo que combina el sistema de almacenamiento cifrado RSA con el estándar GPG y utiliza dos claves generadas, además de una firma para verificar la autenticidad del propietario

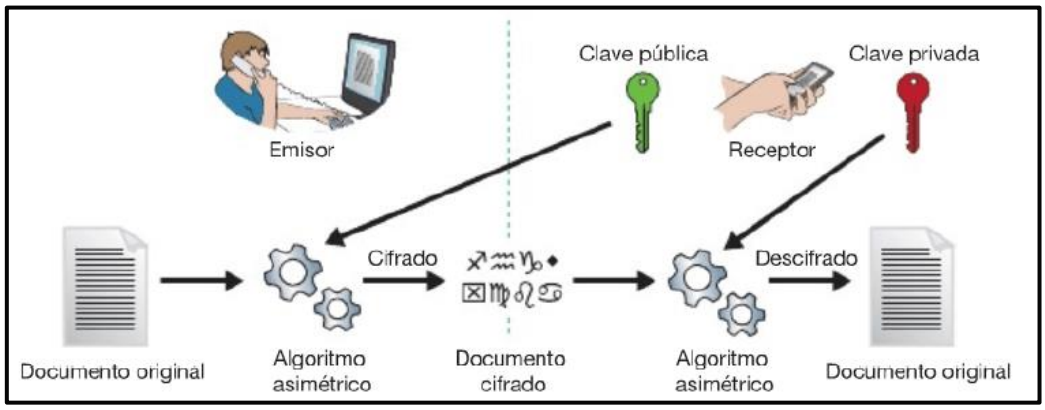
Además, nuestro proyecto incluirá el desarrollo de un sistema de administración de claves públicas para documentos cifrados, que se almacenarán en la nube. Este trabajo aborda temas como la criptografía, el resguardo y la transferencia de datos, además ofrece una solución rentable y eficiente para que las empresas protejan su información crítica o de alto riesgo.

2. CAPITULO 2: MARCO TEORICO

2.1 Cifrado Asimétrico

La utilización de criptografía de llave asimétrica también es crucial para brindar firmas electrónicas, las cuales verifican la autenticidad y la integridad de los usuarios, las transacciones y los datos, desempeñando así un papel central en el mecanismo de seguridad básico de la mayoría de las billeteras criptográficas. (Mendoza, J. C. (2008). Demostración de cifrado asimétrico y simétrico. Ingenio: Revista de Tecnología y Ciencia, (2), 44-50.)

Figura *4*
Proceso de Cifrado Asimétrico



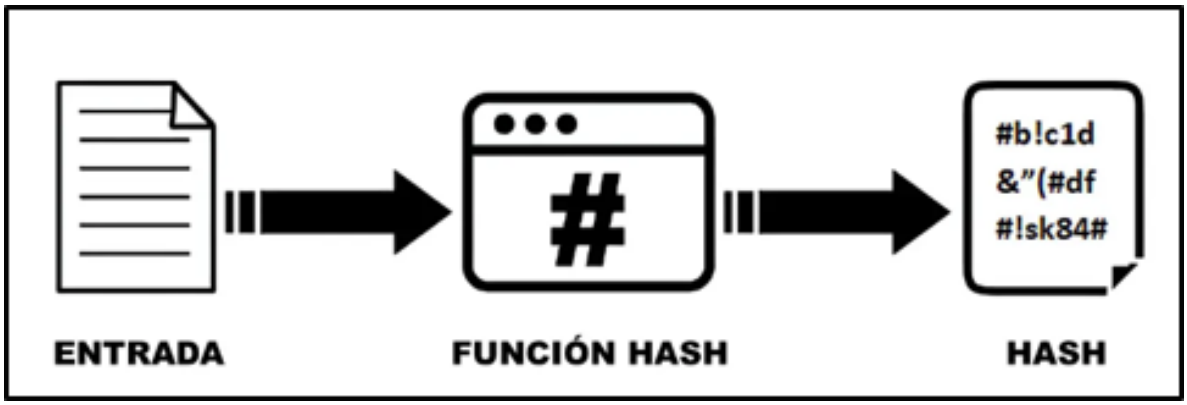
Nota. Representación del proceso de cifrado asimétrico. Adaptado de Criptografía asimétrica, por Jesús Fernández Toledo, 2021, Criptografía asimétrica - Jesús Fernández Toledo

2.2 HASH Funtion

Estas funciones están destinadas principalmente a transformar información en datos codificados que generan una cadena de caracteres única. (Escalona, S. B., & Inclán, L. V. (2011).

Una función hash es el mecanismo que genera la identificación digital única para el mensaje específico. Esta huella digital se crea mediante la reducción del tamaño del mensaje a un tamaño fijo y representándolo como un código. Este código se utiliza para verificar la integridad del mensaje mediante el uso de algoritmos hash comunes y para validar los datos. Algunos algoritmos de funciones hash son MD4, MD5, SHA1, SHA2 y SHA3. Estas funciones se utilizan para proteger la privacidad de las contraseñas. Aunque las contraseñas estén en texto plano y sean accesibles, no pueden ser eliminadas gracias a estas funciones. (Gutiérrez, Y. D., & Lovelle, J. M. C. (2018).

Figura 5
Proceso de Funcionamiento del Algoritmo Hash



Nota: En la figura se presenta el funcionamiento del flujo HASH. Adaptado de Criptotario (2022), "Función Hash". (https://criptotario.com/funcion-hash)

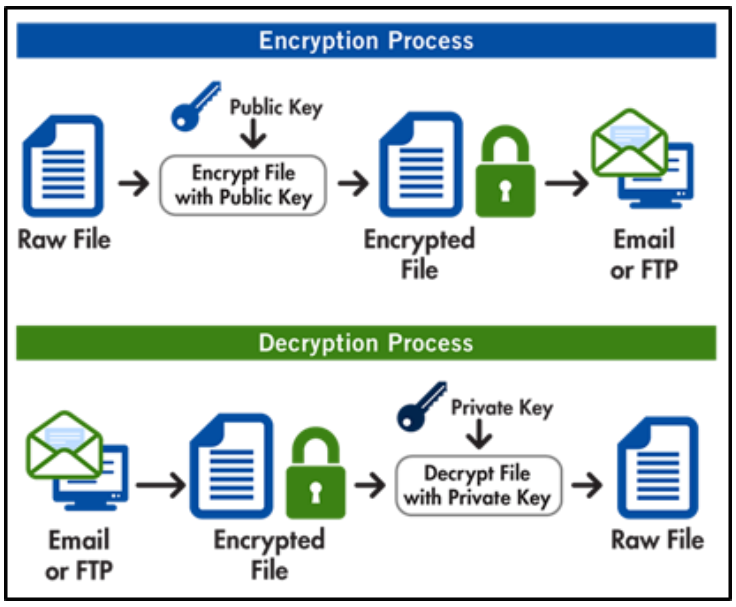
2.3 GPG (GNU Privacy Guard)

Plaza Martín (2021) señala que la encriptación asimétrica es un mecanismo que se utiliza para asegurar la privacidad de archivos y correos electrónicos mediante criptografía de clave pública. Este sistema de cifrado protege las comunicaciones por e-mail, el almacenamiento de archivos y la transferencia de información a través de Internet mediante el uso de una llave pública.

El cual busca el poder encriptar la comunicación entre dos extremos. También, es importante conocer que el uso o aplicación es de recurrir a llaves publicas GPG, justamente para poder realizar una transición de datos de forma cifrada, blindando o asegurando todo dato durante la comunicación.

De acuerdo con Ghasemi (2015), la transmisión cifrada de claves públicas PGP entre destinos contribuye a reforzar la seguridad en los procesos de intercambio de información.

Figura 6
Flujo de Encriptación



Nota. Flujo de encriptación PGP. De PGP, por Deistercloud, 2019, Deistercloud PGP encryption flow documentation

2.4 Algoritmos de Cifrado Asimétrico

En la encriptación de clave compartida, la llave utilizada para el encriptado es similar a la clave utilizada para el descifrado. Por lo tanto, la distribución de claves debe realizarse antes de la transmisión de información. La criptografía simétrica otorga una gran relevancia a la clave en su funcionamiento ya que su seguridad depende directamente de la naturaleza de la clave, es decir, la longitud de la clave, etc.

2.4.1 DES (Data Encryption Standard)

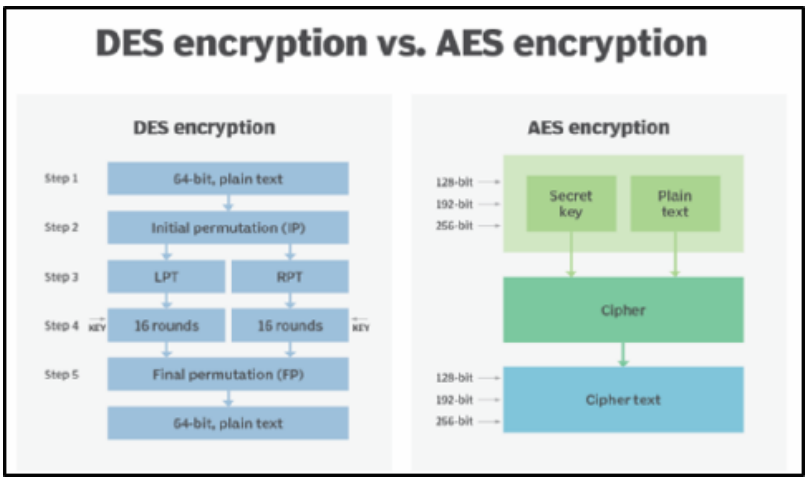
El primer estándar de cifrado difundido oficialmente por el (NIST), organismo encargado del establecimiento de estándares y lineamientos tecnológicos, fue el algoritmo (DES). Este fue desarrollado por IBM a partir de su cifrador Lucifer y adoptado como estándar en 1974. El algoritmo usa una clave de 56 bits y transforma bloques de entrada de 64 bits en bloques de salida del mismo tamaño. Aunque la clave se presenta como de 64 bits, en realidad 8 bits se emplean para paridad impar, por lo que la longitud efectiva es de 56 bits. Con el paso del tiempo, se han identificado diversos ataques y métodos que explotan sus vulnerabilidades, lo que ha llevado a considerar al DES como un cifrado de bloques inseguro (Méndez & Vargas, 2015).

2.4.2 AES (Advanced Encryption Standard)

El algoritmo Rijndael, también conocido como AES (Estándar de Cifrado Avanzado), es un cifrado criptográfico de bloques que tiene la capacidad de cifrar bloques de datos de 128 bits

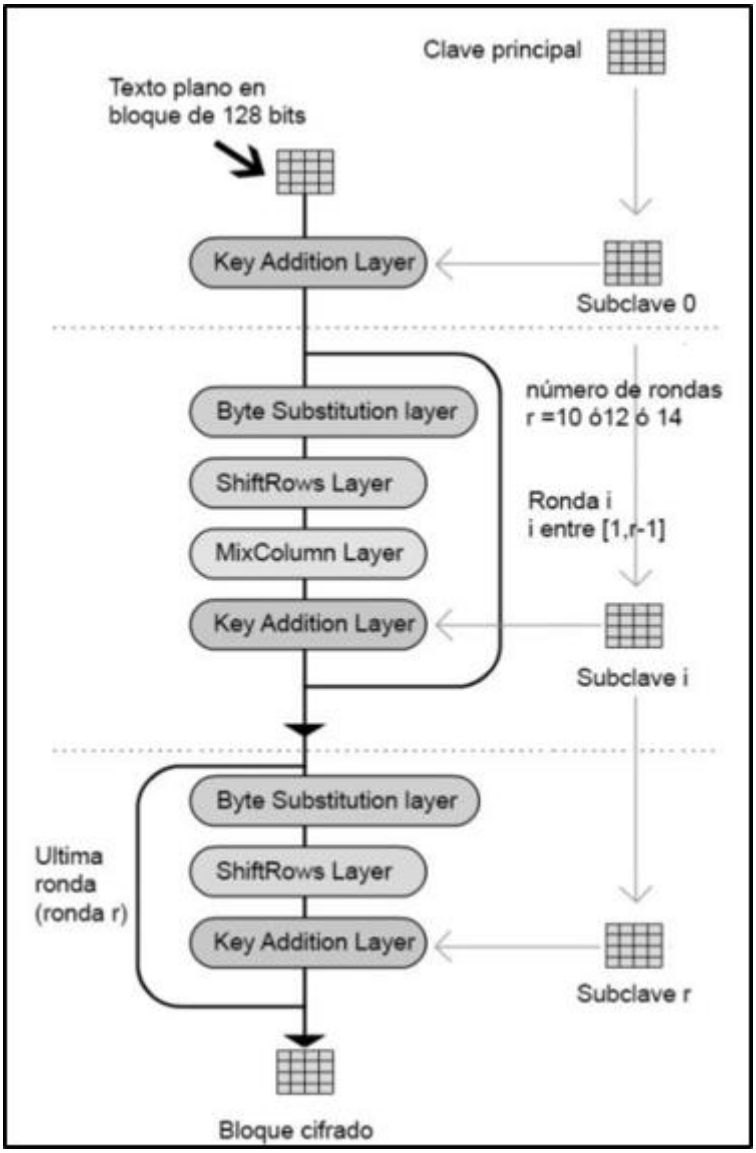
utilizando claves simétricas de 128, 192 o 256 bits. AES fue diseñado como una alternativa al DES. Hasta ahora, el único ataque efectivo conocido contra este algoritmo es el ataque de fuerza bruta. (Mndez, H. A. M., & Vargas, Y. T. M (2015).

Figura 7
Comparación entre los métodos criptográficos DES & AES



Nota. Adaptado de Kryptonsolid (2023). Se observa las diferencias entre DES & AES

Figura 8
Descripción del funcionamiento del algoritmo de cifrado AES



Nota: En el flujograma se ilustra el proceso de funcionamiento del algoritmo de cifrado AES. Adaptado de “Funcionamiento de cifrado AES”, por ResearchGate (2020).

2.5 Algoritmo de Cifrado Asimétrico

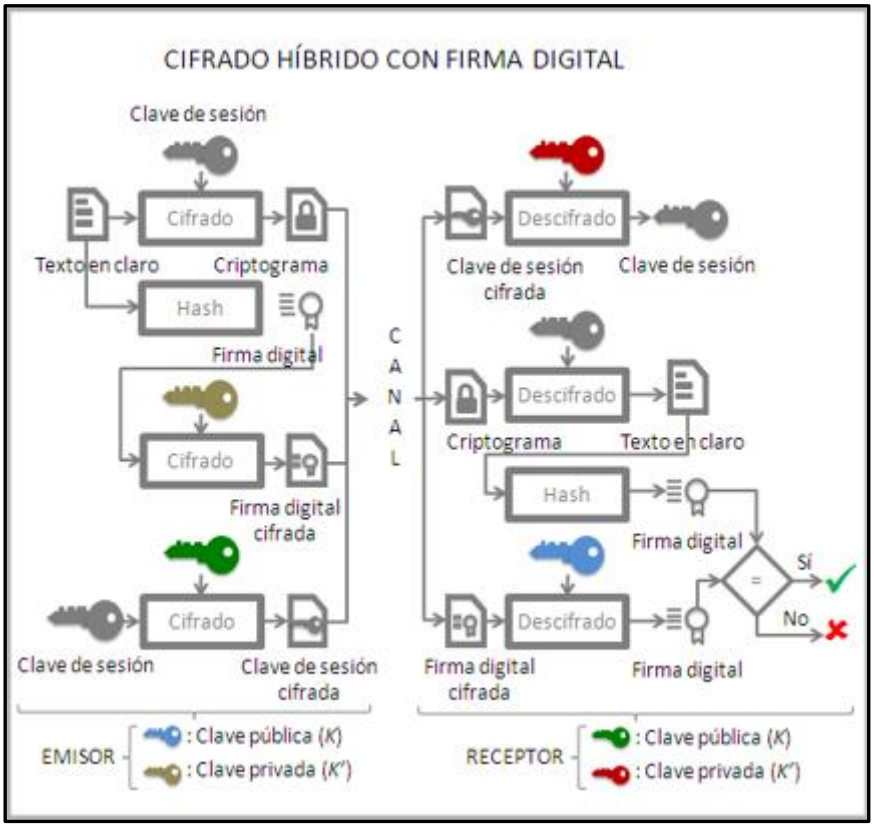
En el cifrado asimétrico, se usan dos llaves diferentes para el cifrado y descifrado: pública y privada. La clave pública está destinada a un uso general, por lo que está disponible para cualquier persona en la red. Cualquiera que quiera cifrar el texto sin formato debe conocer la llave pública del receptor. Únicamente el usuario autorizado puede descifrar el mensaje cifrado mediante su clave privada, la cual se mantiene resguardada y fuera del acceso externo.

2.5.1 RSA (Rivest-Shamir-Adleman)

El criptosistema RSA es un sistema de clave pública seguro y ampliamente utilizado, creado por Ronald Rivest, Adi Shamir y Leonard Adleman. Este fue publicado en 1978 en la revista

Communications of the ACM, donde se presentó como un método para obtener firmas digitales y sistemas de criptografía de clave pública (Rivest et al., 1978). A lo largo del tiempo, el algoritmo RSA ha sido objeto de diversas adaptaciones y mejoras que han permitido su incorporación en múltiples estándares de seguridad. Actualmente, es considerado uno de los métodos de cifrado asimétrico más reconocidos y utilizados en protocolos de seguridad informática como SSL, CDPD y GPG, entre otros. Asimismo, su implementación se ha extendido a numerosas aplicaciones orientadas a la protección y confidencialidad de la información digital (Janampa Patilla, 2020).

Figura 9
Implementación de cifrado híbrido con incorporación de firma digital



Nota: García Larragan, M. (2016). Cifrado Híbrido con firma digital. http://mikelgarcialarragan.blogspot.com/2016/10/criptografia-xxxv-el-algoritmo-rsa-i.html

2.6 Proceso de generación de firmas digitales

El algoritmo de firma digital basado en curvas elípticas (ECDSA) es el análogo de curva elíptica del DSA. ECDSA fue propuesto por primera vez en 1992 por Scott Vanstone en respuesta a la solicitud del NIST (Instituto Nacional de Estándares y Tecnología) de comentarios públicos sobre su primera propuesta para DSS. Fue aceptado en 1998 como

estándar ISO (Organización Internacional de Normalización) (ISO 14888-3), aceptado en 1999 como estándar ANSI (Instituto Nacional Estadounidense de Normalización) (ANSI X9.62) y aceptado en 2000 como estándar IEEE (Instituto of Electrical and Electronics Engineers) estándar (IEEE 1363-2000) y un estándar FIPS (FIPS 186-2).

Los esquemas de firma digital se pueden utilizar para proporcionar los siguientes servicios criptográficos básicos: integridad de los datos, autenticación del origen de los datos (la garantía de que la fuente de los datos es la que se afirma) no repudio (la seguridad de que una entidad no puede negar acciones o compromisos previos).

2.6.1 DSA (Algoritmo de firma digital)

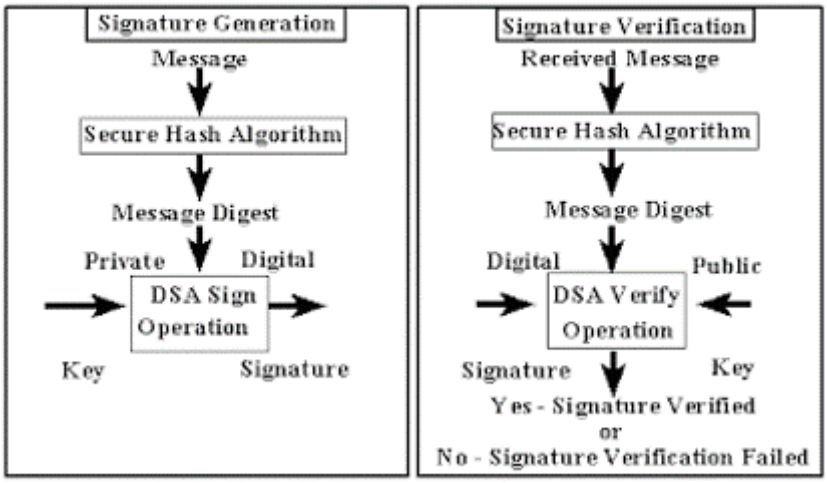
El algoritmo DSA se define como el mecanismo de cifrado mediante clave pública que permite verificar la autenticidad de un mensaje mediante el uso de claves públicas y firmas digitales. Generan un par de llaves públicas y privadas, y utilizar la clave privada para generar firmas de datos.

Este algoritmo utiliza una serie de parámetros, entre ellos:

KG (claves públicas de grupo): Son claves públicas compartidas por un grupo de usuarios.
KU (clave pública): Cada usuario genera su propia clave pública a partir de las claves públicas de grupo.
KP (clave privada): Cada usuario posee una clave privada generada a partir de las claves anteriores.
K (número aleatorio): Se genera un número aleatorio para cada firma digital.
S y R: Son palabras de 160 bits que conforman la firma de un texto específico.
El uso del número K evita la repetición de firmas, lo que contribuye a la seguridad del algoritmo.

De acuerdo con Coloma Macías (2021), el uso de estos parámetros en el algoritmo DSA contribuye significativamente a la protección de la información y a la autenticación segura de los datos.

Figura 10
Proceso de funcionamiento del algoritmo de firma digital DSA



2.7 Estándares de Cifrado

Los estándares de cifrado de datos son un cifrado (método para encriptar información) seleccionado como el Estándar Federal de Procesamiento de Información (FIPS) fue adoptado oficialmente en los Estados Unidos en 1976 y, con el tiempo, alcanzó una amplia aceptación a nivel internacional. Desde su creación, este sistema provocó debate por ciertos elementos de su diseño catalogados como reservados, el uso de una longitud de clave limitada y las dudas respecto a una posible vulnerabilidad intencional atribuida a la Agencia de Seguridad Nacional (NSA). Por ello, fue objeto de un exhaustivo análisis académico, lo que contribuyó significativamente al desarrollo de la comprensión moderna de los cifrados por bloques y sus técnicas de criptoanálisis.

2.7.1 NIST FIPS-140-3

Esta norma especifica qué requisitos de seguridad debe cumplir un producto certificado. Estos requisitos abarcan aplicaciones y áreas enfocadas con la implementación, la seguridad y el diseño seguro de un producto. El estándar proporciona cuatro niveles (Nivel 1 a Nivel 4) de seguridad, cada uno de los cuales aumenta en calidad en comparación con los niveles anteriores.

La validación y las pruebas de procesos se ven muy afectadas al equilibrar el costo de la evaluación con métodos de evaluación sólidos y amplios, buscando activamente vectores de ataque. FIPS 140-3 aplica solo pruebas de estilo de conformidad al verificar si el producto pasa los criterios dados, incluidos en un documento llamado Requisitos de prueba derivados. Hay dos áreas de FIPS 140-3 en las que nos enfocamos: el Programa de validación del módulo criptográfico (CMVP) y Algoritmo criptográfico Programa de Validación (CAVP). CMVP se

refiere a los productos como módulos criptográficos. Cerca de 3.900 módulos están certificados bajo este programa, y a cada uno se le asigna un número de identificación único.

La información sobre cada módulo certificado se puede encontrar en la página de búsqueda.

El programa de validación del módulo criptográfico (CMVP), un esfuerzo conjunto del NIST y el Centro Canadiense para la Seguridad Cibernética (CCCS), valida los módulos criptográficos según FIPS 140-3. El principal objetivo del CMV P es promover el uso de productos que cumplen con un estándar de seguridad. La norma exige que un producto validado cumpla con los requisitos de seguridad derivados de objetivos de seguridad de alto nivel, como implementar correctamente las funciones de seguridad, proteger contra el uso no autorizado, detectar errores y prevenir el compromiso de datos protegidos.

2.8 Protocolo de Correo Electrónico (Email)

En sus primeras etapas, el correo electrónico en Internet fue incorporado como una funcionalidad del protocolo FTP. Posteriormente, en 1980, Suzanne Sluizer y Jon Postel desarrollaron un sistema que más adelante sería denominado SMTP (Simple Mail Transfer Protocol). Este protocolo se sigue utilizando hoy en día (con pequeñas mejoras) para la transmisión de mensajes entre sistemas de Internet.

El protocolo SMTP se diseñó partiendo del supuesto de que los sistemas que intercambiarían mensajes serían enormes ordenadores multiusuario con tiempo compartido que estarían constantemente conectados a Internet. Sin embargo, con la llegada de los PC con acceso intermitente a Internet, se hizo imprescindible una solución para la entrega de correo a estos PC.

2.8.1 SMTP (Simple Mail Transfer Protocol)

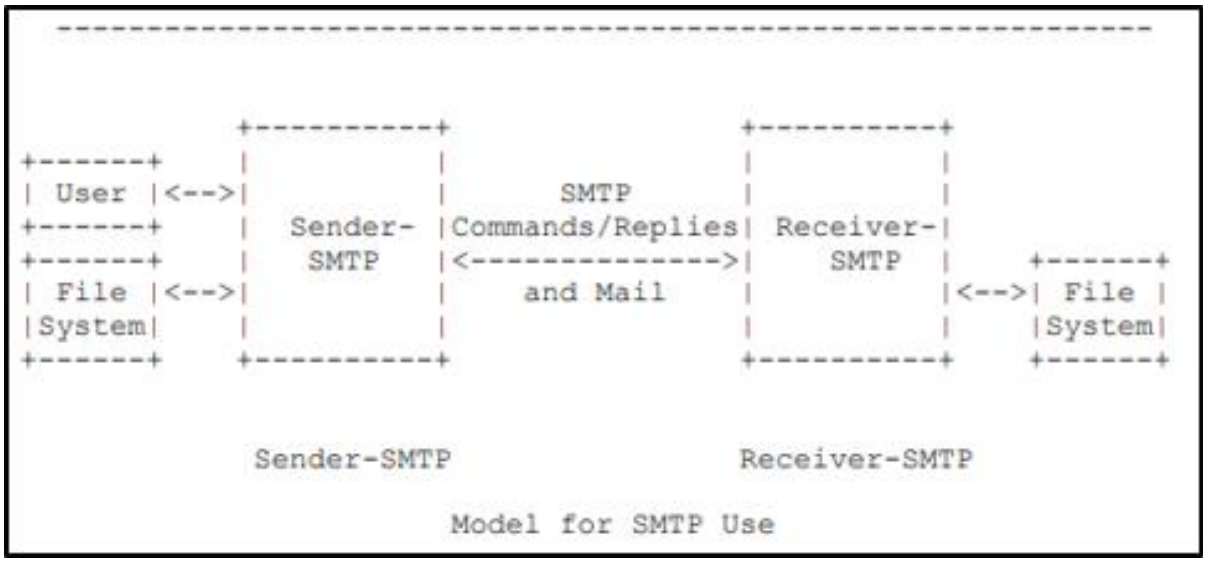
El protocolo SMTP está diseñado para gestionar el envío de correos electrónicos, priorizando la seguridad y eficiencia en su transmisión

La capacidad de reenvío de correo en entornos de servicios de transporte es una característica destacada de SMTP. El servicio de transporte proporciona el entorno de comunicación entre procesos (por sus siglas en ingles IPCE). Es importante entender que los sistemas de transporte (o IPCE) no son lo mismo que las redes. El correo es una aplicación o uso de la comunicación entre procesos. El correo se puede transferir entre procesos en diferentes IPCE mediante la retransmisión a través de un proceso conectado a dos (o más) IPCE.

En el protocolo SMTP, el servidor del remitente envía los comandos hacia el servidor del destinatario, y este último responde mediante mensajes dirigidos nuevamente al servidor emisor como reacción a los comandos recibidos. Si el receptor SMTP esta disponibles, responde con un mensaje de confirmación (OK).

La argumentación del código MAIL es la ruta de retorno que indica de dónde proviene el mensaje. La argumentación del código RCPT es una ruta de reenvío que señala a quién está destinado el mensaje. La ruta de reenvío es la ruta original y La ruta de retorno es el camino que permite reenviar el mensaje al remitente en caso de que ocurra un error durante su retransmisión. Así mismo, se identifica que los comandos y respuestas del servicio o protocolo tienen una sintaxis rígida. *Postel, J. (1982). Rfc0821: Simple mail transfer protocol.*

Figura 11
Modelo de Funcionamiento del Protocolo SMTP



Nota: Se presenta el diagrama de flujo del SMTP, correspondiente al “Model for SMTP Use” de Postel, J. (1982)

2.8.2 Los Procedimientos SMTP

Los procedimientos que se utilizan en SMTP son en varios puntos:

En primer término, se expone el proceso básico del correo electrónico, denominado transacción de correo. Posteriormente, se desarrollan los aspectos relacionados con el reenvío de mensajes, la validación de direcciones de buzones y la ampliación de los grupos de distribución. Asimismo, se abordan los mecanismos de envío hacia terminales en lugar de buzones, así como esquemas combinados, junto con los procedimientos asociados a estas operaciones de apertura

y cierre de sesión. Finalmente, se incluyen observaciones sobre la retransmisión, una nota acerca de los dominios de correo y una discusión sobre la evolución de los roles.

2.8.3 Transacción de Correo SMTP

El proceso de transmisión de mensajes SMTP se compone de tres etapas principales. En primer lugar, la transferencia se inicia mediante el comando MAIL, el cual especifica la identidad del remitente. A continuación, se ejecuta una secuencia de uno o varios comandos RCPT, encargados de definir la información del destinatario. Posteriormente, el comando DATA permite el envío del contenido del mensaje. Finalmente, la operación concluye con la confirmación de la transacción.

El primer paso del procedimiento corresponde al comando MAIL. En este contexto, el campo <reverse-path> contiene la dirección del buzón de origen, y su sintaxis se expresa como: MAIL FROM:<reverse-path> <CRLF>. Este comando indica al servidor SMTP receptor el inicio de una nueva transacción de mensaje, lo que implica el reinicio de su estado interno y de los búferes asociados, incluyendo la información de los destinatarios y los datos del correo. Asimismo, proporciona la ruta inversa que puede emplearse para la notificación de errores. Si la solicitud es aceptada, el servidor responde con el código 250 OK.

Este comando notifica al servidor SMTP receptor el inicio de una nueva transacción de correo, lo que implica la reinicialización de su estado interno y de los búferes asociados, incluyendo la información de destinatarios y los datos del mensaje. Asimismo, establece una ruta inversa que puede utilizarse para la notificación de errores. En caso de ser aceptado, el servidor SMTP responde con el código 250 OK.

Por otro lado, el campo <reverse-path> puede incluir múltiples buzones de correo, constituyendo una lista de origen inverso que identifica tanto los hosts como las direcciones de correo del remitente. En este sentido, el host especificado en <reverse-path> corresponde al sistema que emite el comando.

El segundo paso del procedimiento corresponde al comando RCPT, cuya sintaxis es: RCPT TO:<forward-path> <CRLF>. Este comando especifica la ruta de avance utilizada para identificar al destinatario del mensaje. En caso de ser aceptado, el servidor SMTP responde con el código 250 OK y almacena la ruta de reenvío correspondiente. Sin embargo, si el destinatario no es reconocido, el servidor devuelve un mensaje de error 550. Cabe señalar que este paso puede repetirse tantas veces como sea necesario para incluir múltiples destinatarios.

Por otra parte, el campo <forward-path> puede contener más de un buzón de correo, constituyendo una lista de enrutamiento que incluye tanto los hosts de origen como la dirección del destinatario. En este sentido, el primer host especificado en <forward-path> debe corresponder al sistema que recibe el comando.

El tercer paso del procedimiento corresponde al comando DATA, cuya sintaxis es DATA <CRLF>. Si el comando es aceptado, el servidor SMTP responde con un código intermedio 354, indicando que está listo para recibir el contenido del mensaje. Una vez que el texto del correo es transmitido y almacenado, el servidor confirma la operación mediante la respuesta 250 OK.

Dado que los datos del mensaje se envían a través del canal de transmisión, es necesario señalar explícitamente el final del contenido para poder reanudar el intercambio de comandos y respuestas. En SMTP, esta finalización se indica mediante una línea que contiene únicamente un punto. Para evitar que este símbolo interfiera con el contenido del usuario, se aplica un mecanismo de transparencia de datos.

3. CAPÍTULO 3: ANALISIS DEL PROBLEMA

Se reconoce que hay un problema con la transmisión de información confidencial mediante el email, transferencia de archivos mediante MFT. Todos los servicios de intercambio de información mencionados anteriormente tienen fallas de seguridad y que son trasmitidos por Internet desde la red local de la empresa. Adicional a ello, los factores internos y externos también pueden causar problemas. En otras palabras, las amenazas internas y externas pueden poner en riesgo los datos que los usuarios que intercambian con sus proveedores.

Para realizar el análisis del problema, evaluaremos las amenazas internas y externas identificadas en la descripción del problema. Luego, se realiza una valoración sobre las causas identificadas en la organización y los potenciales resultados de implementar las acciones requeridas para resolver la situación.

Causas del Análisis Interno

Los usuarios no tienen conocimiento o no cuentan con capacitación de seguridad informática.

Hoy por hoy hay más del 70% colaboradores que vienen laborando en diversas áreas de la compañía que maneja información sensible, no tiene o cuenta con alguna noción de lo que se refiere a ciberseguridad. Así como también, no existe una política establecida y avalada por la alta dirección en temas de inducción y fomento de concientización acerca de seguridad informática. A continuación, se adjunta el cuadro con las áreas indicadas.

Tabla 1
Resultados del Proceso de Formación básica en seguridad informática.

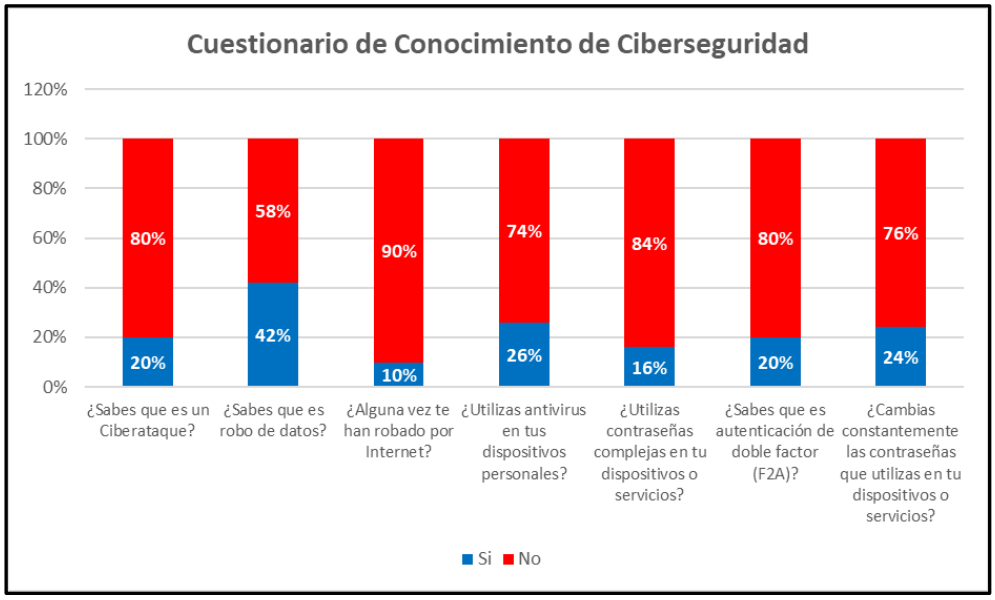
ÁREA ORGANIZACIONAL	INDUCCIÓN SEGURIDAD INFORMATICA
Gestión Comercial	No
Marketing y Comunicación	No
Atención al usuario	No
Gestión Financiera	No
Administración Contable	No
Gestión de Almacenes	No
Logística Operativa	No
Transporte y Distribución	No

Tecnología de la Información	Si

Nota: Resultado de encuesta sobre inducción de Seguridad Informática, por elaboración propia, 2026

De los resultados alcanzados, se evidencia que solo el área de TIC recibe inducción relacionada con seguridad informática. En contraste, las demás áreas de la organización no cuentan con procesos formativos orientados a la concientización y prevención de riesgos informáticos, lo cual podría incrementar la vulnerabilidad frente a incidentes de seguridad.

Figura 12
Cuestionario sobre Ciberseguridad



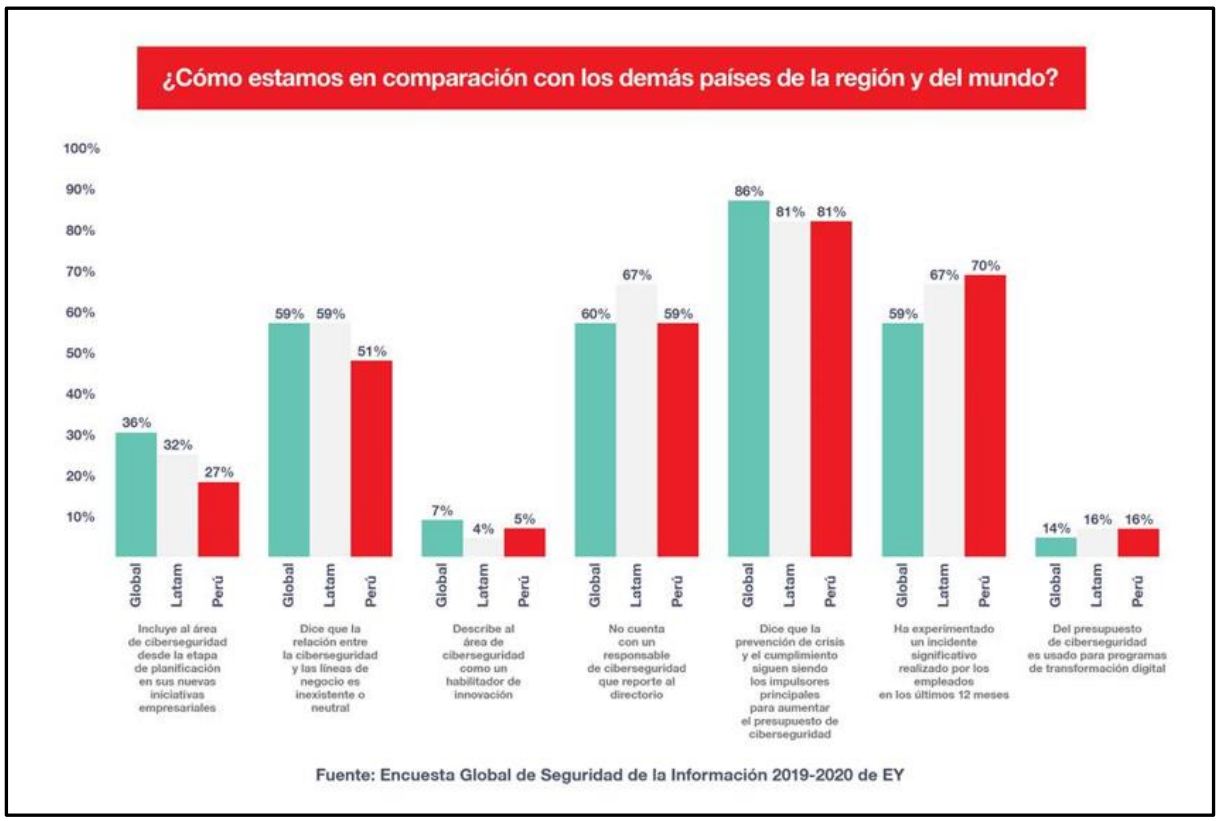
Nota: Cuestionario de conocimiento de Ciberseguridad, por elaboración propia, 2026.

Las organizaciones deben confiar en la tecnología innovadora y disruptiva para ayudarles a expandirse y diferenciarse en una industria. Las organizaciones que no cuentan con controles

de seguridad adecuados para gestionar estos avances tecnológicos pueden enfrentar la pérdida de confianza por parte de los consumidores, las partes interesadas y el mercado.

De acuerdo con un estudio internacional efectuado entre 2019 y 2020 por la consultora Ernst & Young, el Perú figura entre los países con deficiencias en la gestión de la seguridad de la información empresarial. El informe señala que únicamente el 27% de las empresas incorpora la ciberseguridad desde la fase inicial de planificación en sus proyectos de negocio, mientras que el 51% indica que la vinculación entre la ciberseguridad y sus áreas de negocio es nula o poco relevante.

Figura 13
Encuesta Global de Seguridad

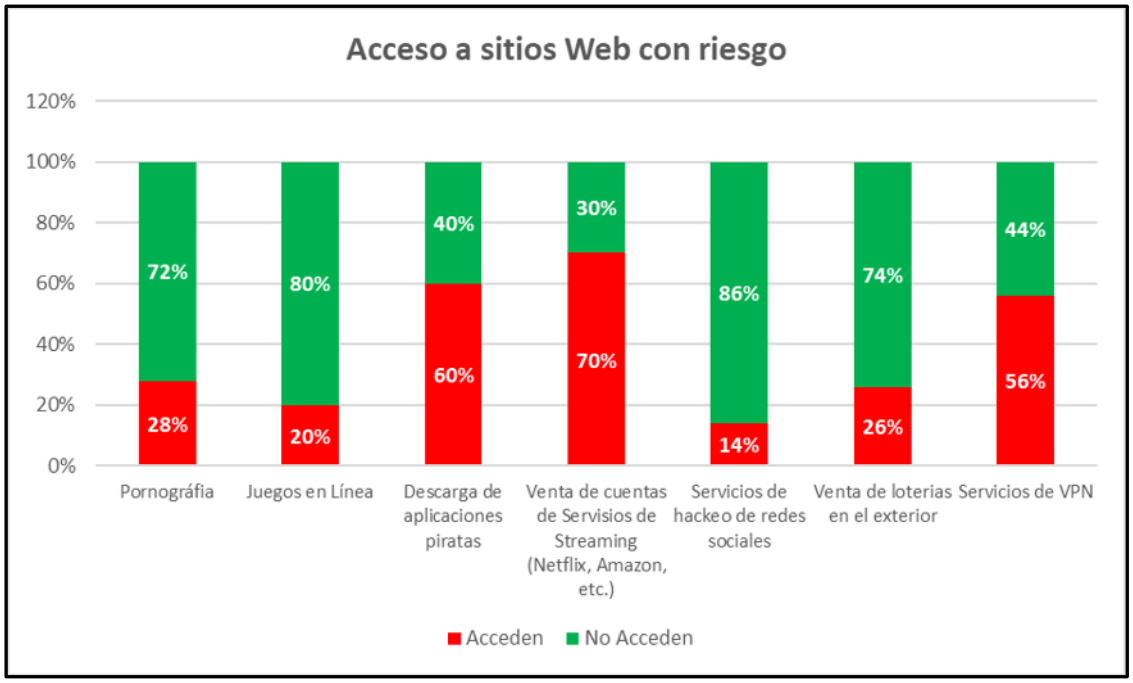


Nota. Encuesta global sobre seguridad de la información. De Encuesta Global de Seguridad de la Información 2019-2020, por Ernst & Young, 2019–2020.

Se ha identificado descuido por parte de los colaboradores en la gestión de información sensible. No utilizan contraseñas seguras para acceder a recursos compartidos y, en algunos casos, comparten sus credenciales con otros.

Se ha encontrado evidencia de que los usuarios acceden a sitios web desconocidos durante la hora del almuerzo, lo que los hace vulnerables a ataques y robo de información.

Figura 14
Tabla descriptiva de los niveles de accesos a sitios web de riesgo elevado



Nota: Porcentaje de acceso a sitios web de riesgos, por elaboración propia, 2026

El servicio de e-mail no tiene controles de cifrado para salvaguardar la información de los mensajes remitidos. Esto significa que los mensajes son vulnerables y pueden ser fácilmente comprometidos. Como se puede ver en la imagen de los encabezados del mensaje, no hay evidencia de que se utilice algún mecanismo o nivel de encriptación.

Figura 15
Cabecera de mensaje de DeusMarket

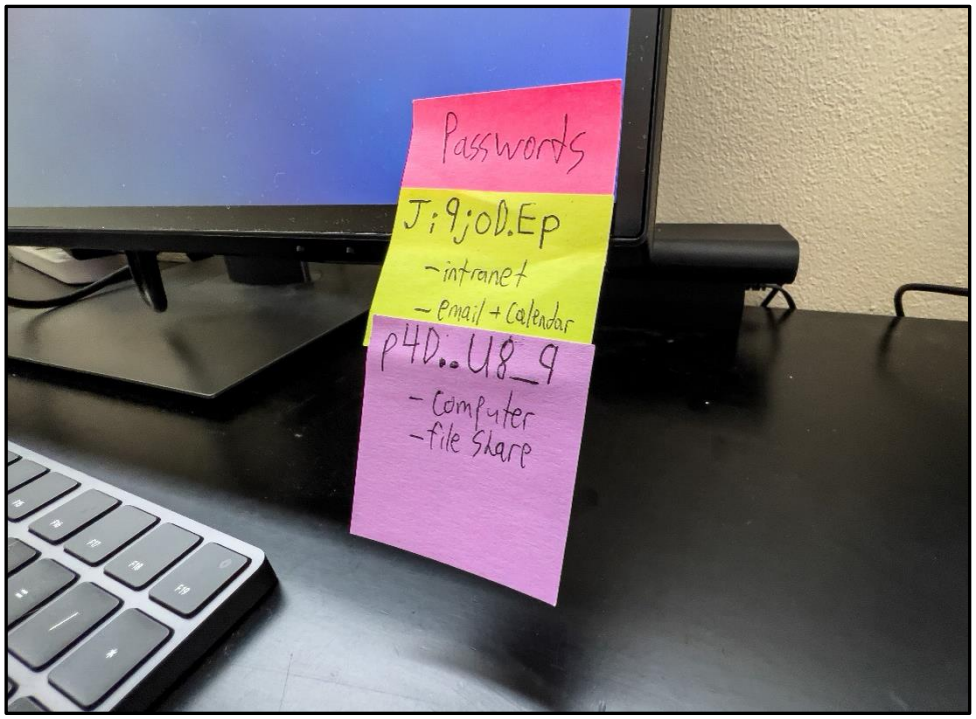


```
Delivered-To: ventasonline@deusmarket.pe
Received: by 10.13.233.66 with SMTP id s63csp689217ywe;
        Sat, 29 Apr 2020 08:03:19 -0700 (PDT)
X-Received: by 10.84.237.8 with SMTP id
s8mr22467927plk.163.1493478199929;
        Sat, 29 Apr 2020 08:03:19 -0700 (PDT)
Return-Path: <sales@xion-pst.cn>
Received: from mta27.xion-pst.cn (mta27.xion-pst.cn.
[67.195.83.90])
        by mx.google.com with ESMTP id
z128si9328326pfz.323.2017.04.29.08.03.19
        for < ventasonline@deusmarket.pe >;
        Sat, 29 Apr 2020 08:03:19 -0700 (PDT)
Received-SPF: pass (google.com: domain of mta27.xion-pst.cn
designates 67.195.83.90 as permitted sender) client-
ip=67.195.83.90;
Authentication-Results: mx.google.com;
       dkim=pass header.i=@xion-pst.cn;
       spf=pass (google.com: domain of mta27.xion-pst.cn designates
67.195.83.90 as permitted sender) smtp.mailfrom= mta27.xion-pst.cn;
       dmarc=pass (p=REJECT sp=REJECT dis=NONE) header.from=xion-
pst.cn Received: from localhost ([127.0.0.1]) by
mta27.communications.yahoo.com (-); Sat, 29 Apr 2017 15:03:05 +0000
X-VirtualServer: sales, mta27.xion-pst.cn, 67.195.83.90
X-VirtualServerGroup: sales
X-MailingID:
50379765::6955760::4826292::377464::ver**1**GUID**VSCDQ56MBQFYCCSSI
7BUYCREEM_17853000001_angel_gv#6h::38872_188814631_0
X-SMHeaderMap: mid="X-MailingID"
X-Destination-ID: ventasonline@deusmarket.pe
X-SMFBL: YW5nZWwubWFydGluLmd1dGllcnJlekBnbWFpbC5jb20=
DKIM-Signature: v=1; a=rsa-sha1; c=simple; d= mta27.xion-pst.cn;
s=201402-std-mrk; i=@xion-pst.cn; h=Content-Transfer-Encoding:
Content-Disposition:Content-Type:Reply-To:MIME-Version: Message-
ID:X-ReportingKey:Subject:Date:To:From; bh=J7Eol8CgtQznS
XENVw+x3+uyP5I=; b=by7eHD3J12ZwHuaDWKQHD18MzsutihYH7QUQTqf87WFYD
LrHkO/D8BX2eIeu8tDQqKygIQ+wI4Su6SKiKNoXPk/OvO2IKKaZWkKjFhfi6dhMN
/OBlqLovZ2kz07cVQ0aOWQMhOZK15jLJyYx4YvMll4n2If14526hUpIZoSZezw=
Content-Transfer-Encoding: quoted-printable
Content-Disposition: inline
Content-Type: text/html; charset="UTF-8"
Reply-To: sales@xion-pst.cn
MIME-Version: 1.0
Message-ID: <50379765.38872@@xion-pst.cn >
X-ReportingKey: LJ2F720V2H0BT1I0C4QKM705SJJSCC-
J404G04M::ventasonline@deusmarket.pe::ver**1**GUID**VSCDQ56MBQFYCCS
SI7BUYCREEM_17853000001_#6h
Subject: Status Sales April
Date: Sat, 29 Apr 2020 15:03:05 +0000
To: ventasonline@deusmarket.pe
From: Sales <sales@xion-pst.cn >
```

Nota: Cabecera de e-mail de DeusMarket, por elaboración propia, 2026

El personal expone sus usuarios y contraseñas de los sistemas en Post-IT's colocado en sus escritorios o lugar de trabajo, dando como resultado una mala práctica de ciberseguridad al no cumplir con una política de Escritorios Limpios.

Figura 16
Evidencia de exposición de passwords



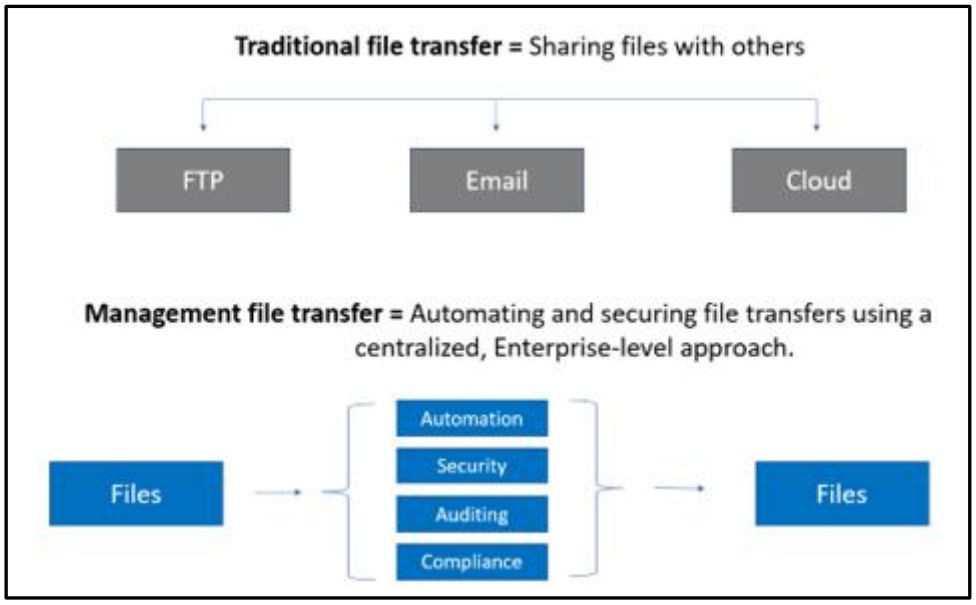
Nota: Se muestra que se encuentra expuestas las contraseñas de usuarios a simple acceso. De "Evidencia de exposición de passwords", Por Elaboración Propia.

El servicio de correo que utiliza la organización "G-suite de Google" brinda una funcionalidad de encriptación básica en formato S/MIME, está orientado a estándares para el cifrado asimétrico que utiliza dos claves vinculadas matemáticamente, una clave pública y una clave privada, pero este cifrado no se cuenta por defecto o no es parte nativa en G-suite, para que se pueda utilizar se debe implementar cargando un certificado que cumpla con los estándares criptográficos y contar con la estructura de criptografía de clave pública (PKCS) #12.

La empresa utiliza la transferencia de información de pedidos a través de MFT (Manager File Transfer). Managed File Transfer, es una tecnología todo en uno. Es una solución segura de transferencia de archivos que cubre todos los aspectos de la transferencia de archivos entrantes y salientes. Las soluciones de MFT utilizan protocolos de red y métodos de encriptación estándar de la industria para proteger los datos confidenciales tanto en tránsito como en reposo. Sin embargo, esta actualmente adolece de mecanismos de encriptación.

Actualmente la compañía realiza transferencias de archivos planos a través de una transferencia de archivos de forma simple sin protección de extremo a extremo, quedando de esta manera propenso a un ataque o robo de información.

Figura 17
Funcionamiento de MFT



- Empresa en constante crecimiento en ventas online
- Riesgos no identificados concerniente a ciberseguridad

Problemas de Análisis Interno

Los servicios para compartir información no cuentan con controles de cifrado para salvaguardar la protección de los datos enviados.

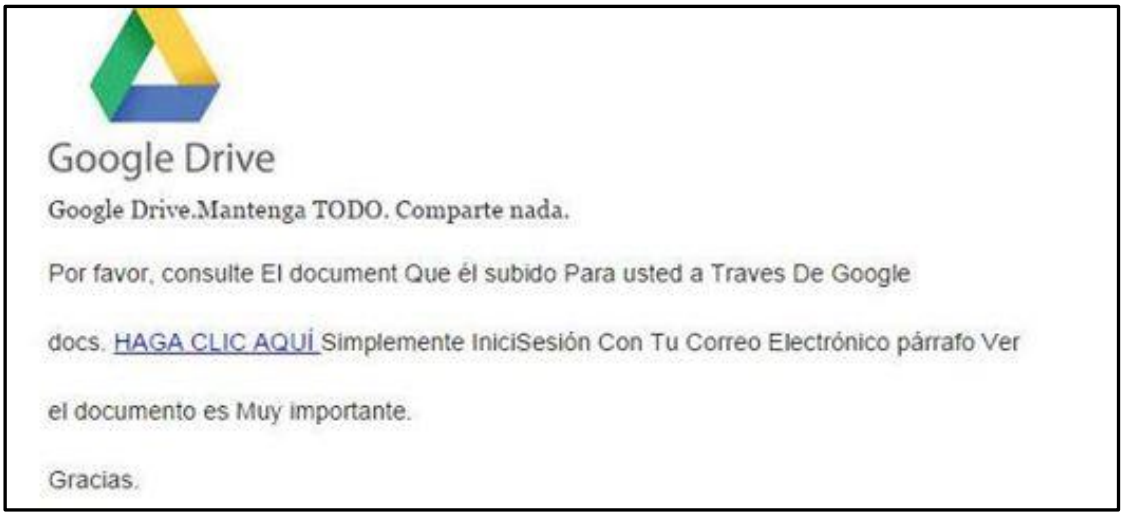
Los usuarios no están informados sobre cómo manejar adecuadamente la información confidencial que reciben.

Consecuencias del Análisis Interno

- Mensajes de correo electrónico de tipo ingeniería social.

El personal está en constante recepción de correos electrónicos maliciosos de tipo ingeniería social, el cual busca el de poder obtener acceso a los recursos compartidos que tiene asignado el colaborador.

Figura 18
Muestra de mensajes maliciosos o engaño

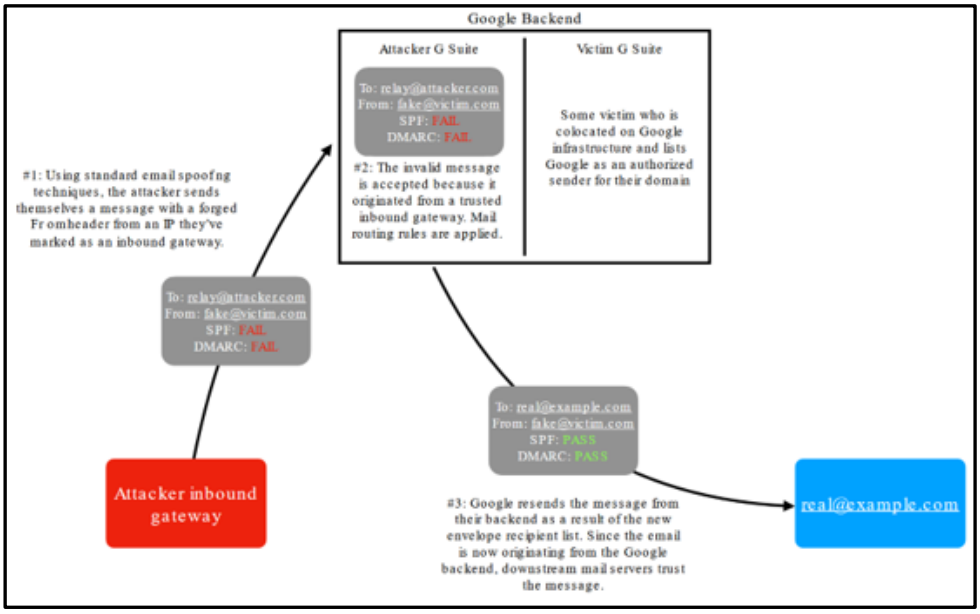


Causas del Análisis Externo

La plataforma de correo electrónico Google G-suite que utiliza la compañía debido a que se acomoda a sus costos y funciones, brinda un cifrado de mensajes muy básicas en formato S/MIME.

Aunque la criptografía de dos llaves, una pública y otra privada, es un estándar, no está implementada de forma nativa en Google G-suite. Para que funcione, es necesario cargar un certificado que cumpla con los estándares de encriptación (PKCS)#12.

Figura 19
Flujo de ciberataque

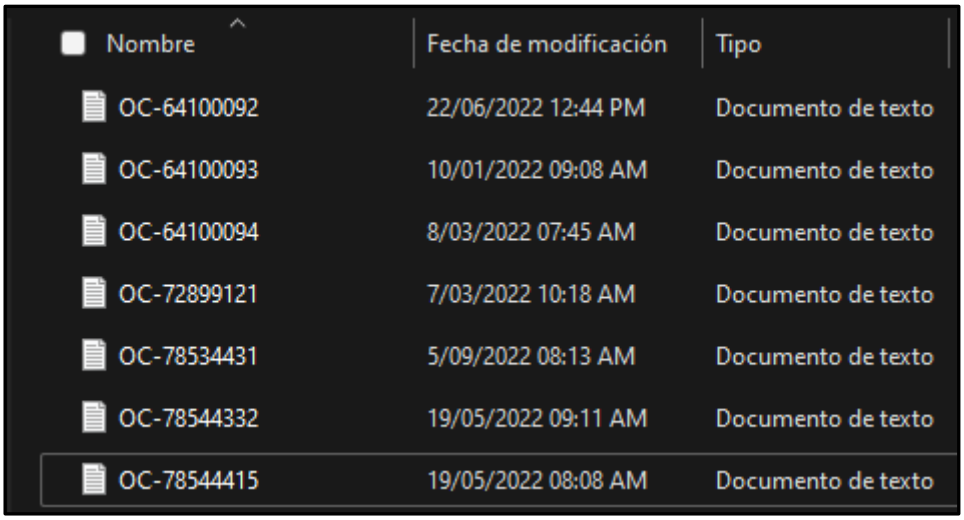


Nota: Se muestra cómo sería un ciberataque al no contar con mecanismos de protección. Por Elaboración Propia.

La organización de acuerdo con las políticas que establece su proveedor de productos utiliza MFT para el intercambio masivos de archivos.

La generación de archivos por lotes de las órdenes de compra o pedidos son generado por la plataforma ERP Odoo, que son descargados en texto plano y enviados al proveedor a través de MFT, el cual es vulnerable ante cualquier posible ataque o intrusión en los servidores donde se almacenan los archivos. (Desarrollar)

Figura 20
Evidencia de archivos en formato plano



Nota: En la figura se evidencia el uso de archivos en formato plano dentro de la plataforma DeusMarket. Elaboración propia

- Información detallada expuesta públicamente sobre los funcionarios de la empresa.

Los datos de los altos ejecutivos de la compañía se encuentran públicamente expuesta, por lo cual, los atacantes pueden recopilar fácilmente esta información y, por lo tanto, realizar ataques de ingeniería social.

Figura 21
Evidencia de exposición de información sensible

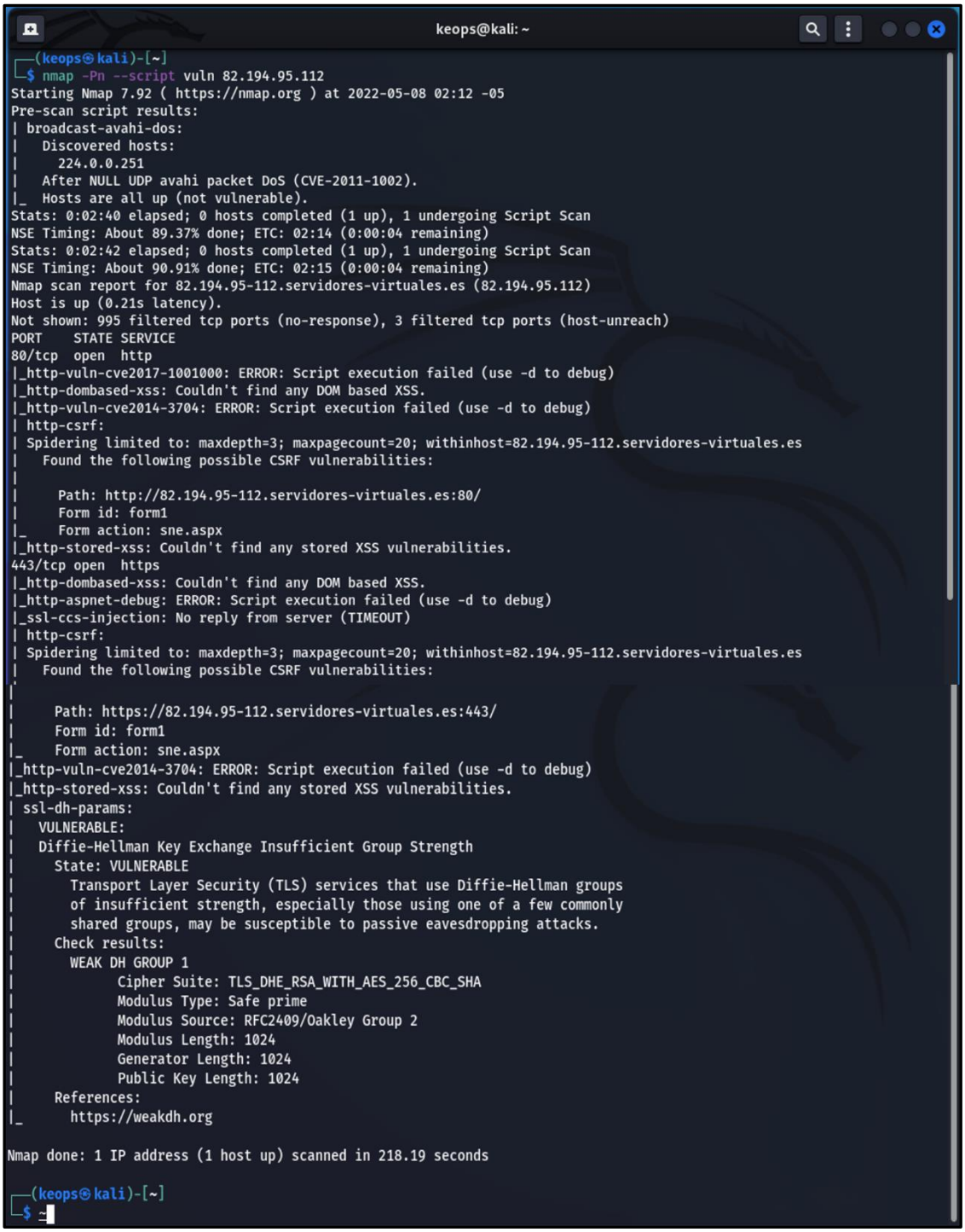


- Riesgos desconocidos en temas de seguridad de la información.

En temas de seguridad de la información, se debe de considerar los riesgos identificados de tipo cero, que aún no es conocido debido a una vulnerabilidad no identificada, ya sea a nivel de aplicación o plataforma que es ejecutada dentro de la infraestructura tecnológica.

Para la reducción de los riesgos y vulnerabilidades del protocolo HTTPS se realiza un escaneo de vulnerabilidades con la herramienta **NMAP**, que nos permitirá detectar las vulnerabilidades en la página web de DeusMarket.

Figura 22
Evidencia de Riesgos y Vulnerabilidades Identificadas.

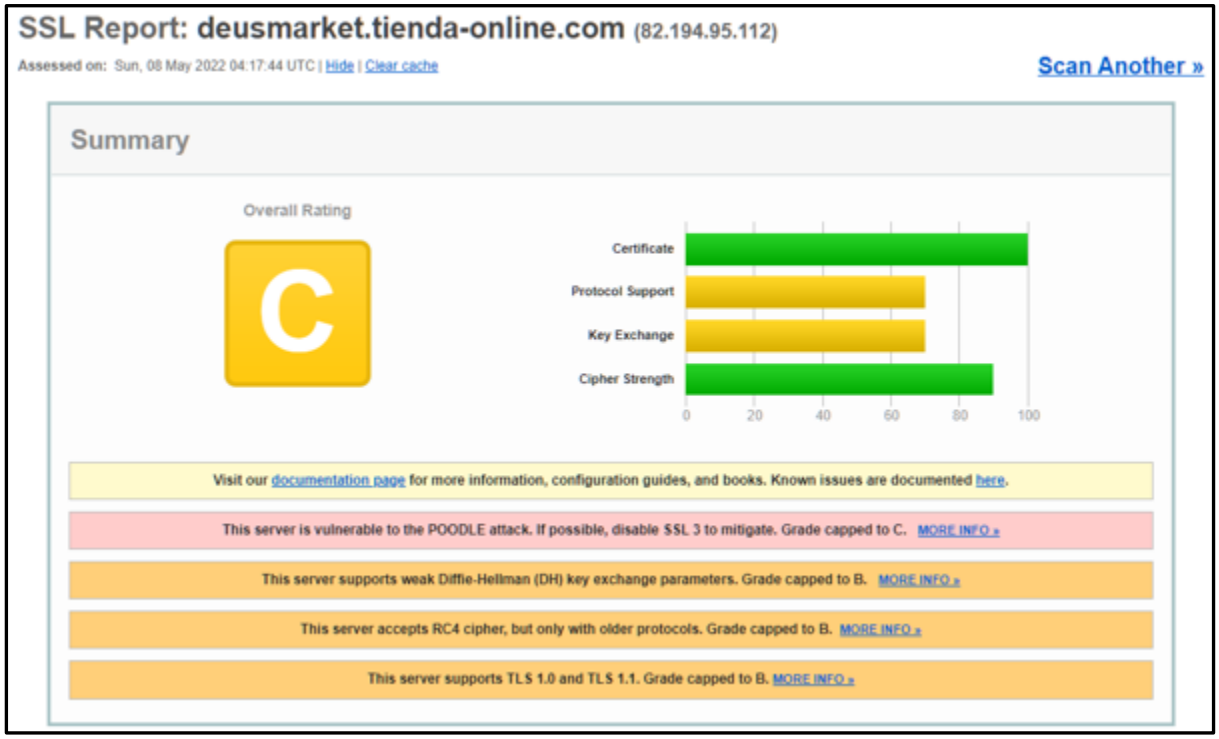


Nota: Se presentan evidencias de los riesgos y vulnerabilidades identificados mediante Nmap en DeusMarket. Fuente: elaboración propia.

Como podemos ver en el reporte de la herramienta NMAP, la página web de DeusMarket presenta vulnerabilidades con el intercambio de llaves de Diffie-Hellman, nos dice que no es lo suficientemente seguro y que es susceptible a ataques de escucha pasiva por ejemplo los Sniffer.

Para la reducción de los riesgos y vulnerabilidades del protocolo TLS/SSL se realiza un escaneo de SSL con la herramienta **QUALYS** el cual nos indicara las vulnerabilidades encontradas en el sitio web corporativo de la empresa DeusMarket.

Figura 23
Informe de evaluación del sitio web DeusMarket



Nota: Informe de evaluación del protocolo SSL de DeusMarket generado por Qualys. Elaboración propia.

Como podemos al realizar el análisis del protocolo SSL de la página DeusMarket, se nota que es vulnerable al Ataque de Man in the Middle del tipo POODLE.

El ataque descrito anteriormente requiere que se establezca una conexión SSL 3.0, por lo que deshabilitar el protocolo SSL 3.0 en el cliente o en el servidor (o ambos) lo evitará por completo. Si cualquiera de los lados admite solo SSL 3.0, entonces se ha ido toda esperanza y se requiere una actualización seria para evitar el cifrado inseguro (Möller, B., Duong, T., & Kotowicz, K. 2014)

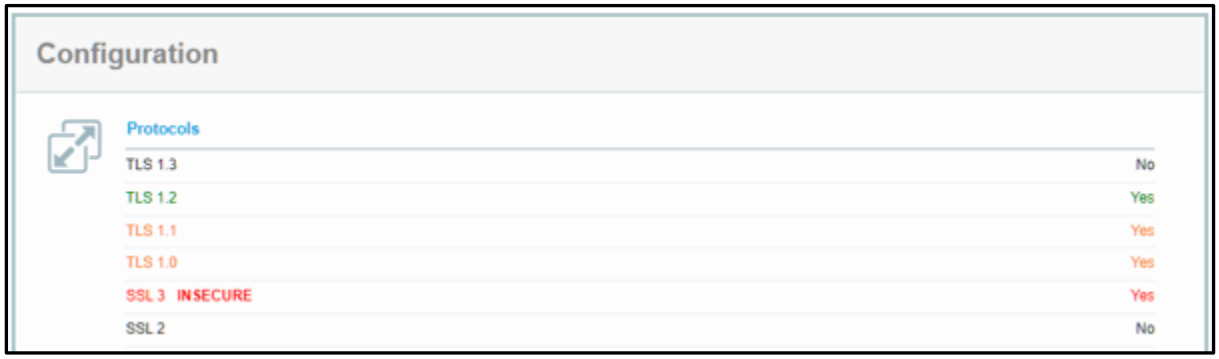
Además, como podemos ver también presenta problemas con el intercambio de llaves Diffie-Hellman (DH), permite que dos partes que inicialmente no comparten información secreta generen una clave secreta mutua (Maurer, U. M., & Wolf, S. (2000).

De igual manera apreciamos que este servidor acepta cifrado RC4, pero solo con protocolos más antiguos, esto nos dice que ese cifrado RC4 es vulnerable al ataque Man in the Middle del tipo BEAST.

El ataque BEAST hizo vulnerable la multiplicidad de cifrados de bloque del algoritmo RC4, como era un algoritmo de flujo, era el único algoritmo conocido que era invulnerable al ataque de tipo BEAST (Ляхов, А. В., & Касьяненко, Н. Г. 2018).

Dentro del reporte que nos brinda la herramienta QUALYS nos muestra los protocolos que ha detectado en la página de DeusMarket.

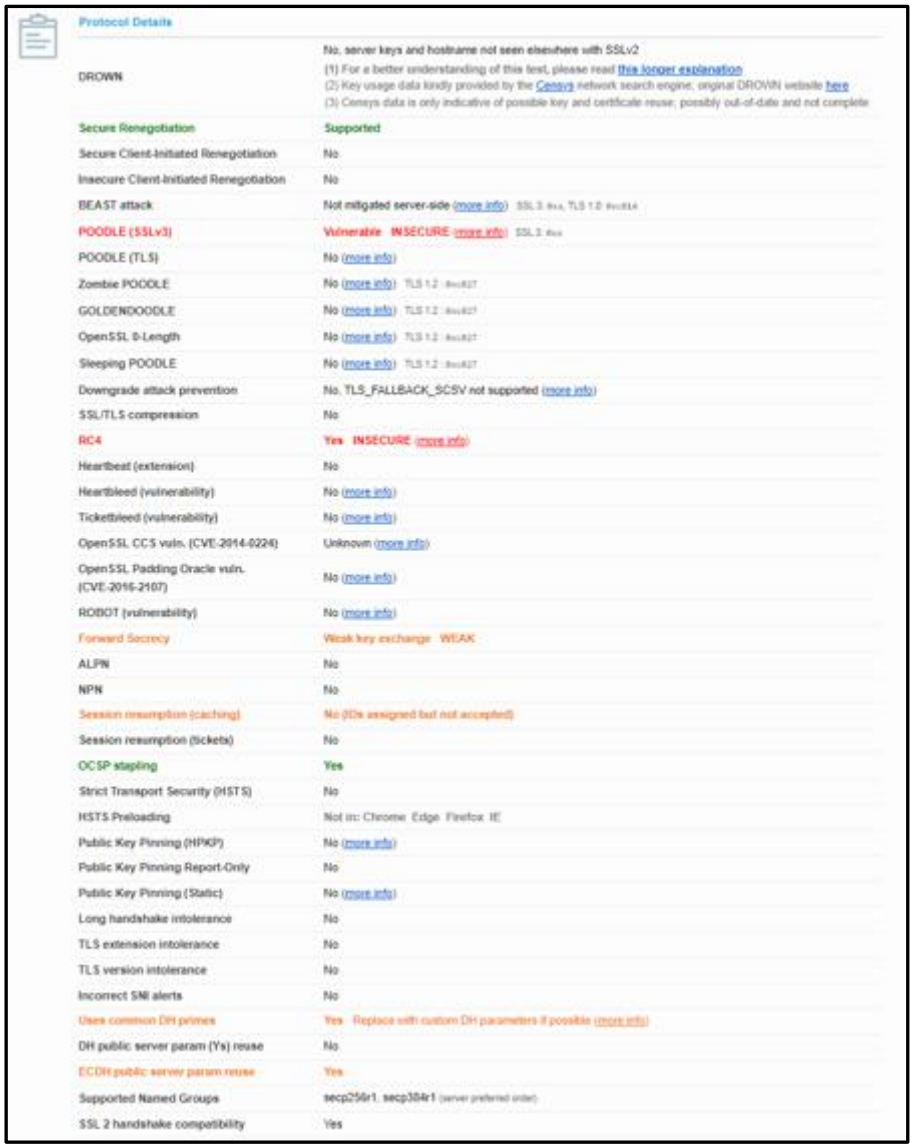
Figura 24
Reporte de los Protocolos Implementados en el Sitio Web DeusMarket



Nota: En la figura se presentan los protocolos identificados en el sitio web de DeusMarket, obtenidos a partir del reporte de seguridad de los protocolos implementados. Elaboración propia.

A continuación, Se presentan detalles ampliados sobre los protocolos implementados en el sitio web DeusMarket.

Figura 25
Detalle de uso de protocoles en el site de DeusMarket

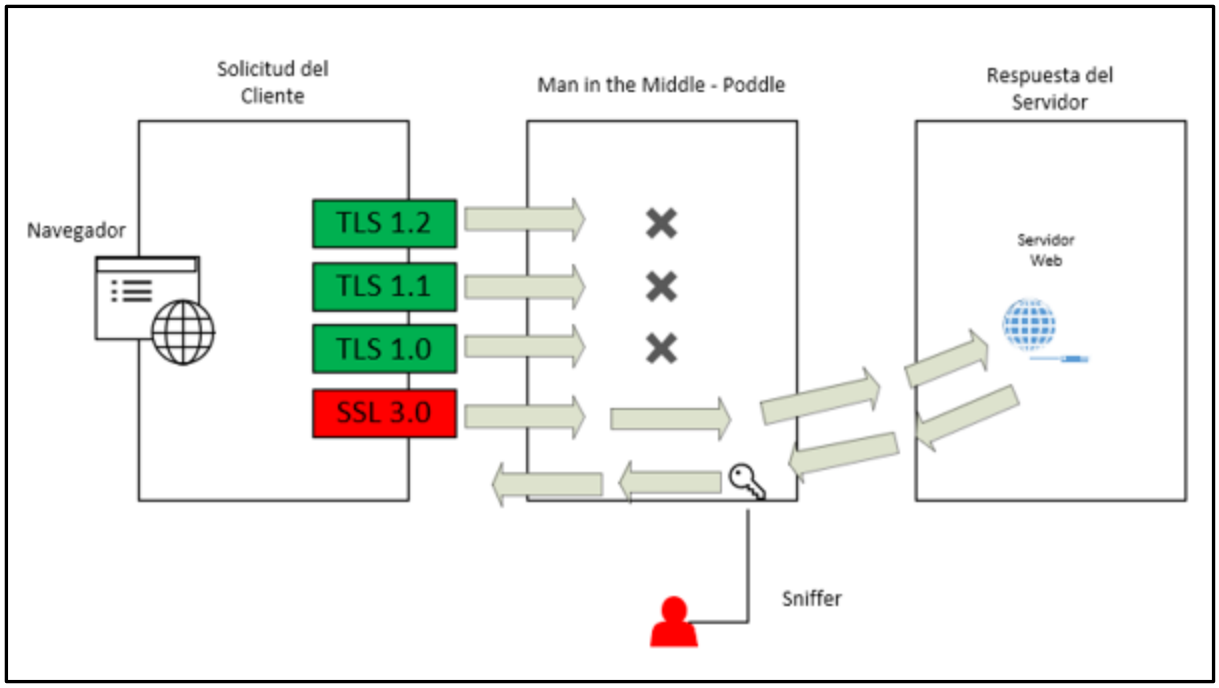


Protocol Details	
DROWN	No, server keys and hostname not seen elsewhere with SSLv2 (1) For a better understanding of this test, please read this longer explanation (2) Key usage data kindly provided by the Censys network search engine, original DROWN website here (3) Censys data is only indicative of possible key and certificate reuse; possibly out-of-date and not complete
Secure Renegotiation	Supported
Secure Client-Initiated Renegotiation	No
Insecure Client-Initiated Renegotiation	No
BEAST attack	Not mitigated server-side (more info) SSL 3: 0xA, TLS 1.0: 0xC014
POODLE (SSLv3)	Vulnerable INSECURE (more info) SSL 3: 0xA
POODLE (TLS)	No (more info)
Zombie POODLE	No (more info) TLS 1.2: 0xC027
GOLDENDOODLE	No (more info) TLS 1.2: 0xC027
OpenSSL 0-Length	No (more info) TLS 1.2: 0xC027
Sleeping POODLE	No (more info) TLS 1.2: 0xC027
Downgrade attack prevention	No, TLS_FALLBACK_SCSV not supported (more info)
SSL/TLS compression	No
RC4	Yes INSECURE (more info)
Heartbeat (extension)	No
Heartbleed (vulnerability)	No (more info)
Ticketbleed (vulnerability)	No (more info)
OpenSSL CCS vuln. (CVE-2014-0224)	Unknown (more info)
OpenSSL Padding Oracle vuln. (CVE-2016-2107)	No (more info)
ROBOT (vulnerability)	No (more info)
Forward Secrecy	Weak key exchange WEAK
ALPN	No
NPN	No
Session resumption (caching)	No (IDs assigned but not accepted)
Session resumption (tickets)	No
OCSP stapling	Yes
Strict Transport Security (HSTS)	No
HSTS Preloading	Not in: Chrome Edge Firefox IE
Public Key Pinning (HPKP)	No (more info)
Public Key Pinning Report-Only	No
Public Key Pinning (Static)	No (more info)
Long handshake intolerance	No
TLS extension intolerance	No
TLS version intolerance	No
Incorrect SNI alerts	No
Uses common DH primes	Yes Replace with custom DH parameters if possible (more info)
DH public server param (Ys) reuse	No
ECDH public server param reuse	Yes
Supported Named Groups	secp256r1, secp384r1 (server preferred order)
SSL 2 handshake compatibility	Yes

Nota: En la figura se detallan los protocolos implementados en la página web de DeusMarket, según el reporte detallado de seguridad realizado. Elaboración propia.

Podemos apreciar cómo se mencionó líneas arriba que la página web de DeusMarket presenta vulnerabilidades del Ataque Man in the Middle de tipo PODDLE en el protocolo SSLv3, además de problemas con el cifrado RC4.

Figura 26
Esquema de ataque Poodle



Nota: En la figura se presenta el esquema correspondiente al ataque POODLE asociado al protocolo SSLv3. Elaboración propia.

Además, podemos ver las carencias que presenta en el paquete de cifrado TLS 1.2

Figura 27
Reporte del Uso de Paquetes de Cifrado.



Nota: Cifrado de paquetes TLS 1.2. De “Cifrado de paquetes en TLS 1.2”, por Elaboración Propia.

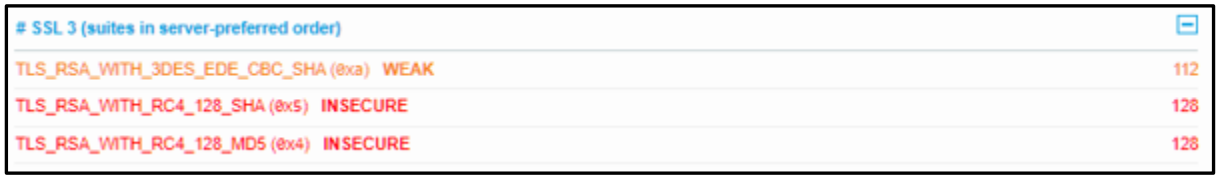
Figura 28
Análisis del Paquete de Cifrado TLS 1.1.



Figura 29
Análisis del Paquete de Cifrado TLS 1.0.



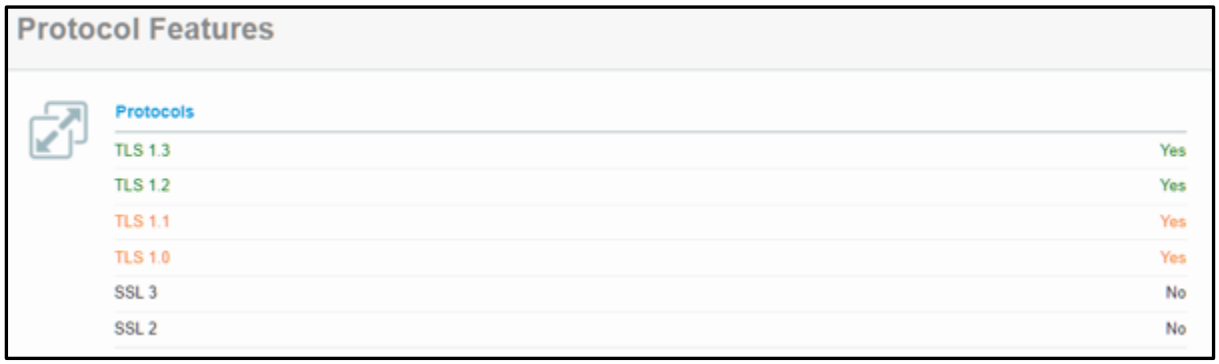
Figura 30
Análisis del Paquete de Cifrado SSL 3.0.



Propuesta de Solución

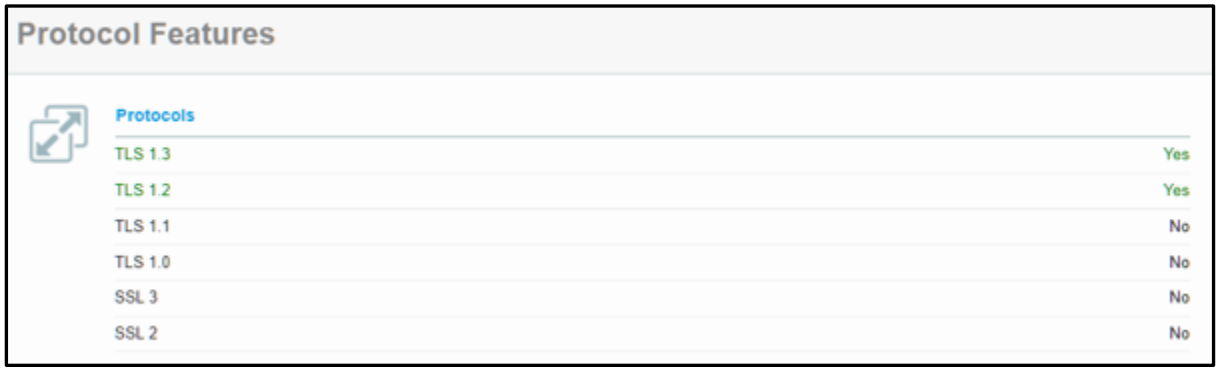
Para la solución de los ataques de Man in the Middle de tipo PODDLE habría que deshabilitar el protocolo SSLv3 y activar la navegación solo con protocolos seguros.

Figura 31
Protocolos TLS luego de la solución para los ataques de tipo PODDLE



Para la solución de los ataques de Man in the Middle de tipo BEAST habría que deshabilitar el protocolo TLS 1.0 y TLS 1.1 y activar la navegación solo con protocolos seguros.

Figura 32
Protocolos TLS luego de la solución



De esta manera la página web de DeusMarket estaría segura de los ataques de tipo BEAST, además que la integridad de los mensajes que se envían será segura.

Para la protección de las transacciones comerciales emplearíamos la criptografía asimétrica.

Efectos de los problemas del Análisis Externo

La plataforma de Google tiene una brecha de seguridad transmitida a través de Internet, ya que no tiene encriptación para proteger la confidencialidad e integridad de los mensajes.
El servicio AS2 solo asegura un canal seguro, pero la información no está cifrada.

Consecuencias de Análisis Externo

El ataque de tipo intermediario (Man in the Middle) puede suceder cuando un ciberdelincuente intercepta y altera la información transmitida desde el cliente hacia los servicios de Google Workspace. Esto puede ocurrir porque el delincuente puede obtener acceso para leer, insertar y modificar los mensajes.

Las cuentas de email de los ejecutivos principales de una empresa pueden ser objeto de ataques. Estos ataques suelen dirigirse a usuarios con poder de decisión y acceso a información confidencial, y pueden tener un gran impacto si se violan las medidas de protección tradicionales. Los ataques suelen realizarse mediante técnicas como el phishing y la ingeniería social.

La vulnerabilidad de los activos de información ante amenazas externas ha aumentado. Los activos sensibles incluyen información confidencial, financiera, estratégica, así como activos tangibles como intangibles (know-how), planes estratégicos y reputación empresarial. También se incluye herramientas de gestión, base de datos, e-commerce, así como sistemas operativos, servidores y dispositivos de comunicación.

Los ataques externos están aumentando en sofisticación, lo que lleva a la necesidad de invertir en equipamiento avanzados de seguridad costosos. El nivel de seguridad existente resulta insuficiente para garantizar la protección de información sensible, y las amenazas presentan un nivel creciente de precisión y complejidad, como los ataques APT (Amenaza Persistente Avanzada), resulta esencial resguardar los datos durante su transmisión hasta el destino final.

Impacto del Problema

Impacto Tecnológico

La consecuencia de tener vulnerabilidades en servicios de transferencia de información con proveedores, como el servicio de e-mail y MFT, puede generar repercusiones en el desarrollo de los procesos tecnológicos de la organización y aumentar el riesgo de ataques. Un ataque

exitoso al servicio de email para obtener acceso podría permitir al atacante acceder a información privada importante en la red corporativa.

Impacto de responsabilidad de la seguridad de la información

La protección de los sistemas y los datos es una prioridad para la organización, de acuerdo con las directrices de seguridad de la información. Se deben considerar factores como la clasificación de la información y las responsabilidades legales y contractuales. En este sentido, es necesario implementar medidas de cifrado como controles para garantizar la seguridad y confidencialidad adecuadas. Un ataque que resulte en la revelación no autorizada de datos confidenciales podría tener graves consecuencias legales y financieras para la empresa, debido a las sanciones establecidas por la regulación vigente. Además, es necesario considerar las directrices sobre el cifrado de archivos y la protección de la ciberseguridad, abarcando el software, el hardware y las redes que puedan representar riesgos en la transmisión de información confidencial.

Impacto Económico

Un incidente de seguridad puede tener un impacto económico significativo debido a los recursos tecnológicos adquiridos y mantenidos por la empresa. La pérdida de información vital puede tener un costo incalculable. Asimismo, la recuperación de la reputación de la empresa después de una fuga de información puede implicar un costo significativo, que podría superar el costo de implementar políticas de seguridad adecuadas.

3.1 Definición de requerimientos

Para llevar a cabo el estudio y la determinación de los requisitos, se debe comenzar por identificar a los interesados de este. Posteriormente, se procederá a definir los requisitos funcionales que se propondrán para este proyecto.

3.2 Interesados del proyecto

Con el fin de asegurar una comprensión clara de los requisitos, primero se debe identificar y documentar a los Stakeholders del proyecto, su nivel de compromiso y autoridad correspondiente.

Tabla 2
Tabla de Interesados del Proyecto

Interesados	Organización	Rol	Compromiso (Baja, Media, Alta)	Autoridad (Baja, Media, Alta)
Gerente de TIC	DeusMarket	Patrocinador	Alto	Alto
Usuarios destinatarios de información	Terceros	Usuarios Proveedores	Bajo	Medio
Jefes	DeusMarket	Usuario	Medio	Medio
Gerentes	DeusMarket	Usuario	Medio	Alto
Gerente General	DeusMarket	Alta Dirección	Alto	Alto
Jefe de Sistemas	DeusMarket	Director del Proyecto	Alto	Alto
Analista de Sistemas	DeusMarket	Miembro del Proyecto	Alto	Medio

3.3 Clasificación de los Interesados

Se representó la información mediante matrices de poder, interés, influencia e impacto utilizando datos de formularios y análisis previos. A continuación, mostramos una tabla para comprender los valores en las matrices y recordar el puesto de cada Stakeholders.

Tabla 3
Clasificación de Interesados

Nombre	Cargo	Representado por
Interesado 1	Gerente General	1
Interesado 2	Gerentes	2
Interesado 3	Jefes	3
Interesado 4	Gerente de TIC	4
Interesado 5	Jefes de Sistemas	5
Interesado 6	Analista de Sistemas	6
Interesado 7	Proveedores	7

3.4 Requerimientos funcionales

A continuación, se presentan los requisitos que la solución propuesta en el proyecto debe cumplir. Estos requisitos son las especificaciones o funciones necesarias para cumplir las expectativas Stakeholders del proyecto. Se proporciona información minuciosa sobre las especificaciones a realizar, aunque no todos los interesados del proyecto necesariamente deben comprender completamente estos requisitos funcionales.

Se presenta una tabla que enumera los requisitos funcionales, los cuales se basan en los problemas previamente definidos. Además, se puede observar que cada requerimiento está asociado a un Stakeholders específico, lo cual indica la prioridad asignada a dicho requisito.

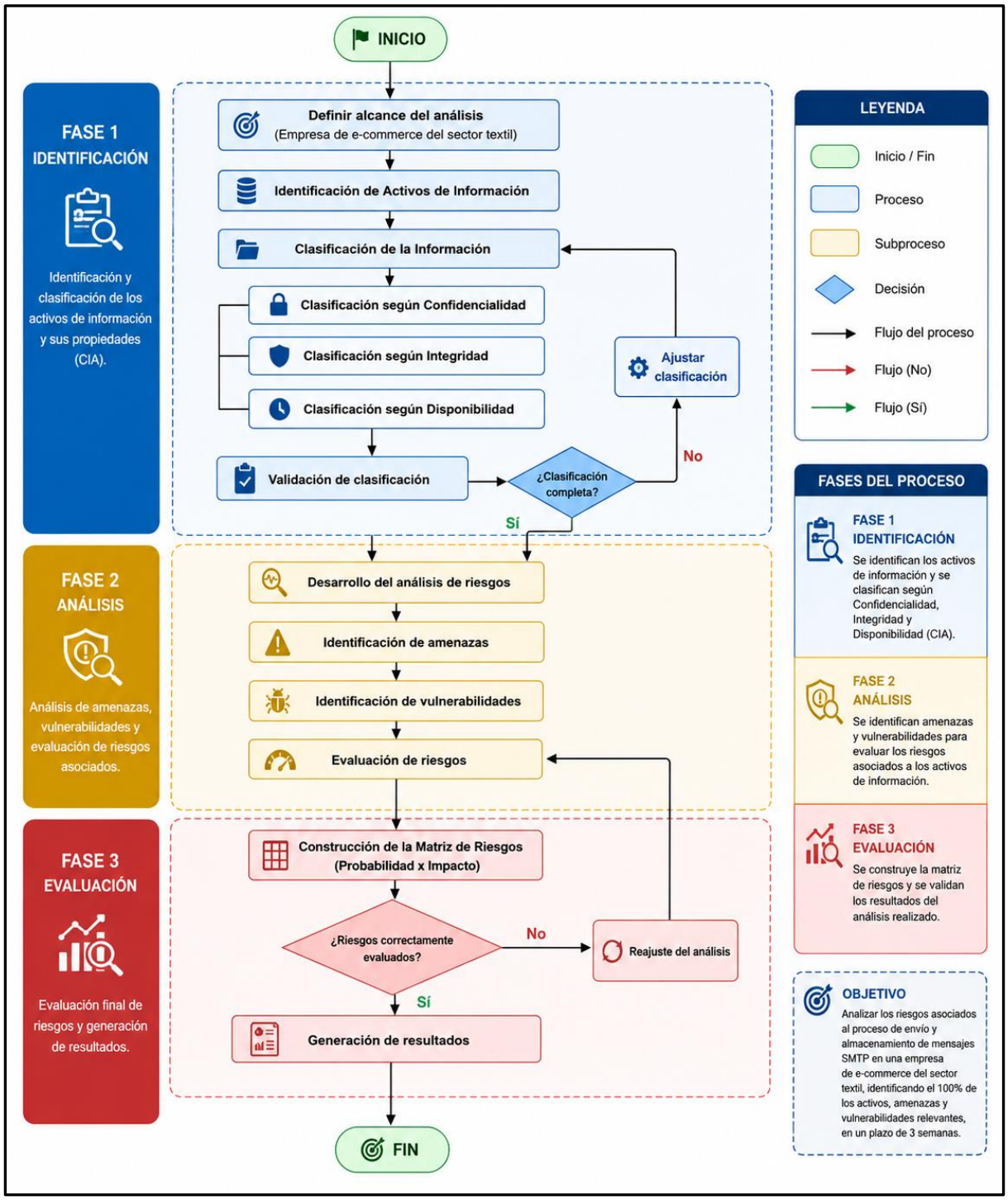
Tabla 4

Tabla de Requerimientos Funcionales

N°	Requerimiento Funcional	Interesados	Prioridad	OE
1	Es necesario aplicar la encriptación a los archivos que han sido clasificados como críticos o confidenciales, los cuales serán enviados únicamente a destinatarios especificados por el remitente.	Jefe de Sistemas, Jefes, Gerentes	Alta	Analizar los riesgos asociados al proceso de envio y almacenamiento de mensajes SMTP en una empresa de e-commerce del sector textil, identificando el 100% de los activos, amenazas y vulnerabilidades relevantes, en un plazo de 3 semanas.
2	Implementar un mecanismo de criptografía asimétrica mediante la generación de claves, con el objetivo de minimizar el riesgo de posibles amenazas que permitan acceder a los datos confidenciales durante el proceso de cifrado y envio.	Jefe de Sistemas	Media	Diseñar el sistema de criptografía asimétrica para el envio seguro de mensajes SMTP, especificando los algoritmos, la gestión de claves y los mecanismos de cifrado y firma digital, en un plazo de 4 semanas.
3	Proponer el cifrado de las operaciones desde el remitente hasta el destinatario. Dando por consiguiente un nivel de seguridad de punto a punto.	Jefe de Sistemas, Jefes, Gerentes y Proveedores	Media	Diseñar el modelo funcional del sistema, incluyendo los procesos de codificación, envio y almacenamiento de mensajes, mediante diagramas y especificaciones técnicas, en un plazo de 4 semanas.
4	Asegurar que toda la información intercambiada por los usuarios, jefes y gerentes se mantenga integra y confidencial, a través de servicios como G-Suite en la nube y AS2.	Jefe de Sistemas	Alta	Diseñar e implementar un sistema automatizado para la codificación por bloques y la transferencia de mensajes mediante AS2, haciendo uso del lenguaje Python, definiendo su arquitectura lógica, flujo de procesamiento y mecanismos de comunicación segura, en un plazo de 4 semanas.

Elaboración Propia

4. CAPÍTULO 4: DISEÑO DE LA SOLUCIÓN



4.1 Analizar los riesgos asociados a los procesos de la empresa

Primeramente, para la identificación de riesgos, nos basaremos en la norma ISO 27005:2013, para lo cual se explicará brevemente de que trata este estándar:

ISO 27005:2013

Las organizaciones internacionales ISO y IEC, dedicadas a la estandarización de normas técnicas, establecieron una colaboración con el objetivo de desarrollar la serie de normas 27000, orientada a la gestión de la seguridad de la información. Dentro de este conjunto normativo se encuentra la norma ISO/IEC 27002, la cual establece un conjunto de buenas prácticas destinadas a facilitar la implementación de un Sistema de Gestión de Seguridad de la Información (SGSI) en las organizaciones (ISO & IEC, 2013).

La norma ISO/IEC 27002 establece una guía detallada para la implementación de controles de seguridad, proporcionando orientaciones para su correcta aplicación dentro de las organizaciones. En este sentido, la selección de dichos controles debe realizarse a partir de un análisis de riesgos enfocado en los activos más relevantes o críticos de la entidad. Asimismo, a diferencia de la percepción común, esta norma puede ser aplicada como soporte para la implementación de un Sistema de Gestión de Seguridad de la Información (SGSI) en organizaciones de naturaleza tanto pública como privada (ISO & IEC, 2013)

El principal propósito de la norma ISO/IEC 27002 consiste en establecer directrices y principios generales que permitan iniciar, implementar, mantener y mejorar de forma continua la gestión de la seguridad de la información dentro de una organización. Asimismo, esta norma contempla la selección, implementación y administración de controles de seguridad, los cuales deben definirse en función del análisis de riesgos asociado al contexto y a los activos críticos de la entidad (ISO & IEC, 2013).

Finalmente, los controles del cual nos guiaremos serán 2 del dominio 8, de la norma ISO 27002:2013:

RESPONSABILIDAD DEL ACTIVO

Esta fase tiene como objetivo identificar los activos de información y establecer las responsabilidades asociadas a cada uno, con el fin de definir medidas de protección adecuadas en función de un análisis de riesgos previamente realizado.

ESTRUCTURA DE LOS ACTIVOS SEGÚN SU CLASIFICACIÓN

El propósito de esta etapa consiste en identificar los activos de información y establecer los responsables correspondientes, permitiendo evaluar y aplicar mecanismos de protección acordes con el nivel de riesgo asociado a cada activo.

Finalmente, se procederá con la implementación de los lineamientos definidos en los incisos 4.1.1, 4.1.2 y 4.1.3.

PROCESO METODOLÓGICO PARA LA IDENTIFICACIÓN E INVENTARIO DE ACTIVOS DE LA INFORMACIÓN

A lo largo del presente trabajo se propone la implementación de una metodología orientada a la adecuada gestión del inventario de activos de información. Dicho proceso contempla actividades de identificación, clasificación y registro de los activos, desarrollándose a través de cuatro etapas consideradas fundamentales:

DEFINICIÓN:

La etapa de definición comprende la selección de los activos de información que serán incorporados al inventario institucional. Para ello, es necesaria la intervención de un equipo encargado de la gestión y supervisión de los recursos de información que posee la organización, administrados por los responsables de cada área.

CONTROL:

La revisión consiste en evaluar la permanencia de un activo dentro del inventario, así como verificar si los valores asignados durante el proceso de clasificación requieren modificaciones. De manera general, el inventario de activos podrá ser validado o revisado en cualquier momento a solicitud del responsable del proceso correspondiente.

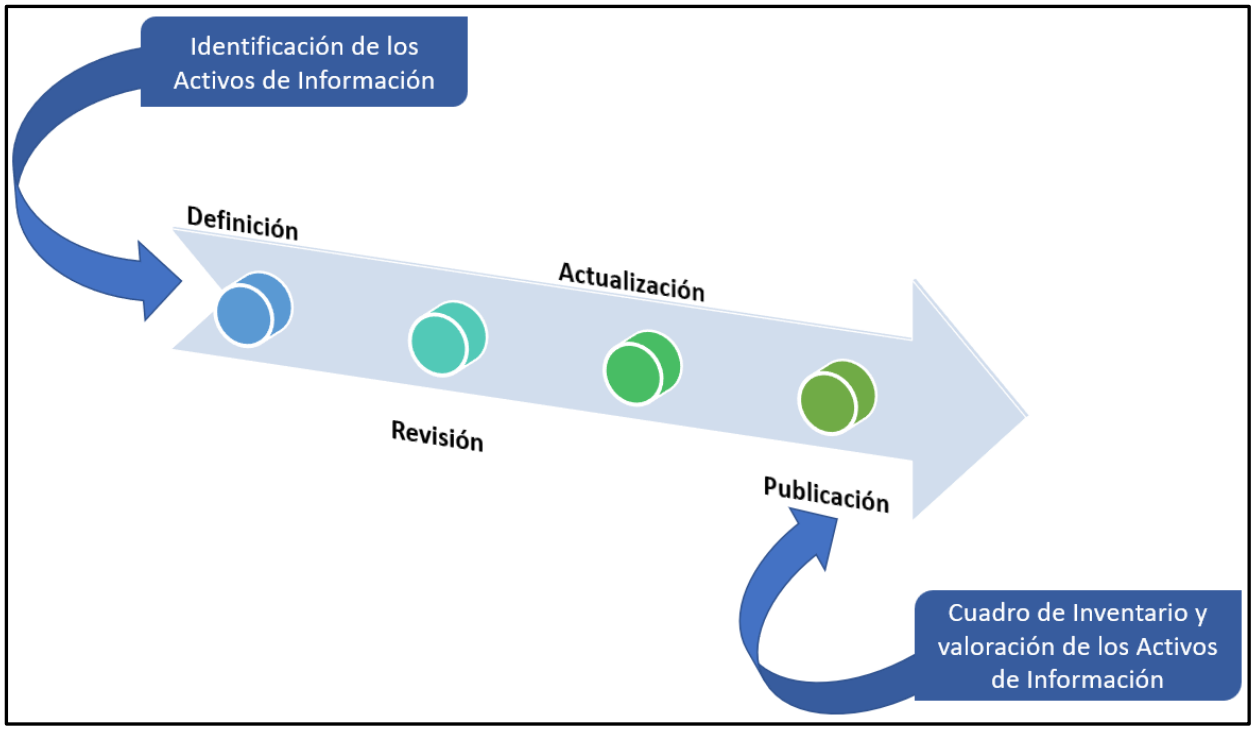
ACTUALIZACIÓN:

Una vez identificadas las modificaciones necesarias dentro del inventario, cada área o proceso deberá efectuar las actualizaciones respectivas en el registro de activos de información, garantizando así la vigencia y confiabilidad de la información documentada.

PUBLICACIÓN:

El registro de activos de información debe ser considerado como un documento de carácter “Confidencial” y no debe permitir su edición por parte de los usuarios autorizados. La modificación de este estará restringida exclusivamente al responsable del proceso, previa aprobación del oficial de seguridad de la información o su equivalente.

Figura 33
Ciclo de Gestión para el Inventario de Activos de Información.



Nota: Ciclo del inventario de activos de información. Elaboración propia.

4.1.1 Identificación de Activos de Información

En este proyecto se establecen las pautas fundamentales para implementar el registro de los activos de información administrados por DEUS MARKET. El propósito es identificar los activos disponibles, definir su adecuado uso, así como establecer los roles y responsabilidades correspondientes existen en relación con ellos y qué nivel de clasificación debe asignar a cada recurso. Se presentarán métodos y directrices específicas para el adecuado tratamiento de los activos. Además, se definen tres niveles (Alto, Medio y Bajo) que permiten valorar globalmente los activos en la entidad, con el fin de identificar aquellos que deben tener un tratamiento prioritario.

4.1.1.1 Clasificación de la Información

Antes de encriptar la información guardada, es importante clasificarla para asegurar que se le brinde la protección adecuada. Esto se hace teniendo en cuenta su valor, criticidad y características específicas, y se debe aplicar un correcto manejo. La clasificación se basará en áreas de la organización, nivel de usuarios, funcionamiento interno y cumplimiento de estándares como el 27001:2013, ISO 27002 y ISO 27005.

4.1.1.2 Clasificación de acuerdo con la Confidencialidad

El acceso a la información por parte de usuarios, grupos o procesos autorizados debe definirse en función de su rol o perfil, conforme a las políticas y directrices internas establecidas.

4.1.1.3 Clasificación de acuerdo con la Integridad

Conforme a lo establecido en la ISO/IEC 27002, la integridad de la información se entiende como la propiedad que asegura que los datos se mantengan precisos y libres de alteraciones no autorizadas durante todo su ciclo de vida, desde su creación hasta su disposición final.

4.1.1.4 Clasificación de acuerdo con la Disponibilidad

En esta categoría, la información debe permanecer disponible y permitir su actualización por parte de usuarios autorizados cuando se requiera. Además, siguiendo el mismo criterio de la clasificación por integridad, se establecen tres niveles: alto, medio y bajo.

Tabla 5
Clasificación de los activos en función de su proceso y subproceso

Código	Proceso	Sub-Proceso	Activo	Descripción	Ubicación	Dueño	Custodio	Categoría	Clasificación
GTIC-001	Pedido a proveedores	Envío de pedido	Servicio de envió de pedido	Servicio de envío de pedidos a través de MFT	Lógica	Infraestructura de TI	Jefe de Infraestructura TI	Aplicación	Confidencial
GTIC-002	Pedido a proveedores	Generación de archivo de pedid	Archivo de pedido	Generador de archivo de pedido en ERP Odoo	Lógica	Infraestructura de TI	Jefe de Infraestructura TI	Aplicación	Confidencial
GTIC-003	Servicio de Email	Servicio de Email	Servicio de Email	Servicio de mensajería por Google G-Suite	Lógica	Infraestructura de TI	Jefe de Infraestructura TI	Aplicación	Confidencial
GTIC-004	Servicio de AD	Servicio de AD	Servidor AD	Servicio de Gestión de usuarios	Lógica	Infraestructura de TI	Jefe de Infraestructura TI	Aplicación	Confidencial
GTIC-005	Servicio de ERP	Servicio de ERP	Servidor de ERP	Servidor ERP Odoo	Lógica	Infraestructura de TI	Jefe de Infraestructura TI	Aplicación	Confidencial

Tabla 6
Clasificación de activos en función de su nivel de criticidad

		Criterio						
Código	**Sub-Proceso**	**Confidencialidad**	**Integridad**	**Disponibilidad**	**Trazabilidad**	**Clasificación**	**Criticidad**	**Nivel de Criticidad**
GTIC-001	Envió de pedido	100	100	100	100	100	100	Extremo
GTIC-002	Generación de archivo de pedido	100	100	100	100	100	100	Extremo
GTIC-003	Servicio de Email	100	100	100	100	100	100	Extremo
GTIC-004	Servicio de AD	100	100	100	50	50	80	Alto
GTIC-005	Servicio de ERP	100	100	100	100	50	80	Alto

Tabla 7

Clasificación de activos según su tipo

CATEGORIA	TIPO
Aplicación	Un sistema de información o cualquier fuente de información que se centre en el procesamiento y la gestión automáticos de datos e información.
Terceros	Cualquier empresa ajena a la sociedad que posea información sobre el Organismos de Evaluación y Fiscalización.
Digital	Cualquier objeto que almacena datos en formato digital o electrónico.
Física	Cualquier recurso que albergue datos guardados de forma física, como registros de personal, contratos, hojas de impresión, entre otros.

Tabla 8

Clasificación de activos en función de su categoría de uso

USO	DESCRIPCIÓN
Público	Todos los datos de acceso público en cumplimiento con la legislación N° 27806, conocida como la Ley de Transparencia y Acceso a la Información. - Cualquier información de acceso público sin restricción
Confidencial	Toda la información es confidencial bajo la legislación N° 27806 - Ley de Transparencia y Acceso a la Información Pública. - Se refiere a aquellos datos que contienen información personal y cuya divulgación podría vulnerar la privacidad y el ámbito personal y familiar. - Se refiere a datos que incluyen sugerencias, consejos u opiniones generados durante el proceso de deliberación y consulta previa a la toma de decisiones gubernamentales, siempre y cuando dicha información esté disponible. - Se refiere a datos que conciernen a investigaciones en curso relacionadas con la aplicación de los poderes legales otorgados a la administración estatal. - Se trata de datos creados o recibidos por asesores legales o abogados que forman parte de los órganos de la gestión pública.

Tabla 9
Clasificación de activos de acuerdo con su nivel de impacto

Criterio	Bajo (10)	Medio (50)	Alto (100)
Confidencialidad	La información que se encuentra en los activos se categoriza como de carácter público.	La información contenida en un activo que contiene información de identificación personal es confidencial.	La información contenida en un activo se clasifica como reservada o confidencial.
Disponibilidad	Si un activo no está disponible, esto no causa ninguna repercusión en los objetivos ni en la continuidad del proceso.	La falta de disponibilidad de un activo durante un período inferior a un día puede comprometer los objetivos y la continuidad de la operación.	Es necesario que el activo esté permanentemente disponible, ya que su ausencia podría afectar tanto los objetivos como la continuidad de la operación.
Integridad	Los daños o cambios no autorizados no afectarán el propósito o la toma de decisiones del proceso.	Los daños o cambios no autorizados afectarán el propósito o el proceso de toma de decisiones.	Cualquier alteración o cambio no autorizado tendrá consecuencias en los resultados o en la toma de decisiones del proceso.
Trazabilidad	El activo en cuestión no posee información personal y no está involucrado en una actividad significativa que demande tener un seguimiento detallado.	El activo en cuestión carece de información personal y no está involucrado en una actividad de importancia que demande un seguimiento preciso.	El activo posee información personal y/o está involucrado en una actividad de gran importancia que demanda un seguimiento minucioso a nivel de transacciones.
Clasificación	Recurso de datos categorizado como de acceso público.	Recurso de datos categorizado como de carácter confidencial.	Recurso de datos categorizado como de acceso restringido.

Tabla Clasificación de activos de acuerdo con su impacto. Fuente: Elaboración propia

4.1.2 Desarrollo del análisis de riesgos

Se trata de un proceso estructurado orientado a la identificación, valoración y evaluación de posibles amenazas asociadas a un proyecto, proceso o actividad. Su finalidad principal es reconocer dichos riesgos, estimar tanto su probabilidad de ocurrencia como su nivel de impacto, y establecer medidas que permitan reducirlos o eliminarlos de manera efectiva.

De manera general, este proceso comprende las siguientes etapas:

A. Identificación de peligros potenciales:

Consiste en reconocer y registrar los posibles eventos o situaciones que podrían representar un riesgo para el desarrollo del proyecto o actividad.

B. Evaluación de riesgos:

Consiste en evaluar la posibilidad de que cada riesgo identificado llegue a materializarse, además de establecer el nivel de impacto o las consecuencias que este podría ocasionar dentro del contexto del proyecto o proceso.

C. Definición de medidas de mitigación:

Se establecen acciones y estrategias enfocadas en reducir tanto la posibilidad de ocurrencia como las consecuencias derivadas de los riesgos previamente identificados.

D. Ejecución y seguimiento de las medidas:

Se implementan las acciones definidas y se realiza un monitoreo continuo con el fin de verificar su eficacia y asegurar su correcta aplicación.

El análisis de riesgos es importante porque ayuda a identificar los riesgos antes de que ocurran, lo que permite a los responsables del proyecto o actividad tomar medidas preventivas y mitigar impactos no esperados en el proyecto o actividad. Además, también ayuda a optimizar la toma de decisiones y asegurar la calidad del proyecto.

Para este proyecto se utilizará el siguiente Análisis de impacto

Figura 34

Tabla de evaluación del impacto cualitativo.

Análisis Cualitativo :

Escala de medida cualitativa de la probabilidad	
Categoría	Definición de la Categoría
Alto	Es muy probable la materialización del riesgo o se presume que llegará a materializarse.
Medio	Es probable la materialización del riesgo o se presume que posiblemente se podrá materializar.
Bajo	Es poco probable la materialización del riesgo o se presume que no llegará a materializarse.

Análisis Cuantitativo :

Probabilidad	Nivel
1	Bajo
2	Medio
3	Alto

Escala de medida caulitativa de impacto	
Categoría	Definición de la Categoría
Alto	Si el hecho llegara a presentarse, tendría alto impacto o efecto sobre la entidad.
Medio	Si el hecho llegara a presentarse, tendría medio impacto o efecto sobre la entidad.
Bajo	Si el hecho llegara a presentarse, tendría bajo impacto o efecto sobre la entidad.

Impacto	Nivel
1	Bajo
2	Medio
3	Alto

Figura 35

Matriz de evaluación cualitativa del impacto.

Análisis Cualitativo :

Puntaje	Nivel del Riesgo	Descripción
1	Aceptable	Riesgo insignificante. No se requiere ninguna acción.
2	Tolerable	Menores efectos que pueden ser fácilmente remediados. Se administra con procedimientos rutinarios
3	Moderado	Debe ser administrado con procedimientos normales de control.
4	Moderado	Debe ser administrado con procedimientos normales de control.
6	Importante	Se requiere atención de la alta dirección. Planes de tratamiento requeridos, implementados y reportados a los jefes de las oficinas, divisiones, entre otros.
9	Inaceptable	Se requiere acción inmediata. Planes de tratamiento requeridos, implementados y reportados a la Alta Dirección.

			Impacto		
			Bajo	Medio	Alto
			1	2	3
Probabilidad	Alto	3	3	6	9
	Medio	2	2	4	6
	Bajo	1	1	2	3

4.1.3 Matriz de Riesgo

En el marco de la gestión de riesgos, la matriz de riesgos se utiliza como un instrumento que permite reconocer, examinar y valorar las amenazas o vulnerabilidades que pueden impactar a una organización. A partir de este análisis, la toma de decisión se orienta a la aplicación de controles y estrategias de mitigación apropiadas.

En la matriz de riesgos, la probabilidad se clasifica típicamente en una escala de baja a alta, mientras que el impacto se clasifica en una escala de bajo a alto. Cada riesgo se ubica en la matriz de acuerdo con su probabilidad e impacto estimados, lo que permite identificar rápidamente los riesgos más críticos que deben ser gestionados.

La matriz de riesgos se aplica como apoyo para la toma de decisiones y para la gestión de procesos, planificar acciones para reducir o eliminar los riesgos identificados. También puede ser útil para monitorear y controlar los riesgos a medida que se implementan las estrategias de mitigación.

Para este proyecto se utilizará la siguiente matriz de riesgo.

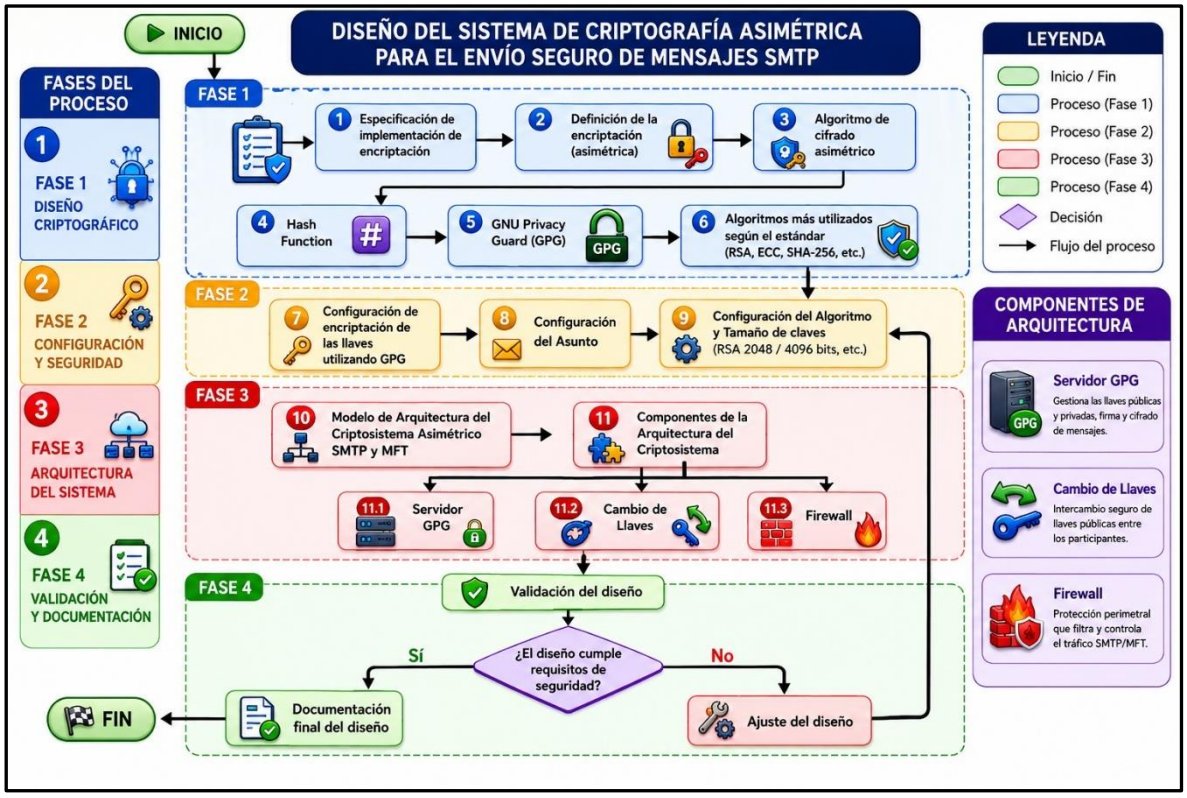
Ítem	DEPENDENCIA	PROCESO	SUB PROCESO	OBJETIVOS DEL PROCESO	ACTIVOS DE INFORMACIÓN (SUB PROCESO)
1	Gerencia de Compras	Pedido a proveedores	Envio de pedido	Servicio de envio de pedidos a traves de MFT	Todos
2	Gerencia de TIC y Procesos	Servicio de Email	Servicio de Email	Servicio de mensajeria por Googel G-Suite de forma segura	Todos
3	Gerencia de Compras	Pedido a proveedores	Generación de archivo de pedido	Generador de archivo de pedido en ERP Odoo	Todos
4	Gerencia de TIC y Procesos	Servicio de AD	Servicio de AD	Servicio de Gestión de usuarios	Todos
5	Gerencia de TIC y Procesos	Servicio de ERP	Servicio de ERP	Servidor ERP Odoo	Todos

IDENTIFICACIÓN DEL RIESGO							Riesgo Inherente			
Cod. Riesgo	**Descripción del Riesgo**	**Propietario del Riesgo**	**Tipo de Riesgo (Estratégico / Operativo / Financiero / Cumplimiento / Corrupción / Tecnología / Seguridad de la Información)**	**Causas Internas del Riesgo**	**Causas Externas del Riesgo**	**Efectos / Consecuencias**	**Probabilidad (P)**	**Impacto (I)**	**P * I**	**Nivel del Riesgo (Inaceptable / Importante / Moderado / Tolerable / Aceptable)**
GTIC-001	Existe el riesgo de que, debido a configuraciones inadecuadas del sistema MFT y posibles ataques externos como Man-in-the-Middle, los pedidos enviados sean interceptados o accedidos por terceros no autorizados, lo que podría generar la filtración de información sensible y afectar la confidencialidad de los datos y la reputación de la organización.	Infraestructura de TI	Financiero	Configuración incorrecta del sistema MFT Falta de cifrado en la transmisión de archivos Deficiente gestión de accesos y permisos	Ataques de interceptación (Man-in-the-Middle) Intentos de intrusión por parte de hackers Uso de redes externas no seguras	Pérdida de confidencialidad de la información Filtración de datos de proveedores Daño reputacional	3	3	9	Inaceptable
GTIC-003	Existe el riesgo de que, debido a contraseñas débiles, falta de autenticación multifactor y ataques externos como phishing o robo de credenciales, se produzca el acceso no autorizado a cuentas de correo electrónico, lo que podría comprometer la confidencialidad de la información y permitir el uso indebido de cuentas institucionales.	Infraestructura de TI	Operativo	Uso de contraseñas débiles Falta de implementación de MFA (autenticación multifactor) Deficiente gestión de accesos	Ataques de phishing Robo de credenciales Ataques de fuerza bruta	Acceso no autorizado a correos Fuga de información sensible Suplantación de identidad	3	3	9	Inaceptable
GTIC-002	Existe el riesgo de que, debido a errores de configuración, datos desactualizados o fallas en integraciones externas, se generen archivos de pedido incorrectos en el sistema ERP, lo que podría ocasionar pedidos erróneos, afectar el proceso de compras y generar pérdidas económicas.	Infraestructura de TI	Financiero	Configuración incorrecta del ERP Datos maestros desactualizados Errores en la parametrización del sistema	Cambios no informados por proveedores Fallas en integraciones externas Información inconsistente de terceros	Generación de pedidos incorrectos Retrasos en el proceso de compras Pérdidas económicas	2	3	6	Importante
GTIC-004	Existe el riesgo de que, debido a contraseñas débiles, falta de controles de autenticación y ataques externos como fuerza bruta o phishing, se produzca el acceso no autorizado a cuentas del servicio de directorio activo, lo que podría comprometer la confidencialidad, integridad y disponibilidad de los sistemas de la organización.	Infraestructura de TI	Corrupción	Contraseñas débiles o mal gestionadas Falta de políticas de seguridad Ausencia de autenticación multifactor (MFA)	Ataques de fuerza bruta Robo de credenciales Phishing	Acceso indebido a sistemas Fuga de información Compromiso de la seguridad organizacional	2	3	6	Importante
GTIC-005	Existe el riesgo de que, debido a una gestión inadecuada de usuarios, contraseñas débiles y ataques externos como fuerza bruta o phishing, se produzca el acceso no autorizado al sistema ERP, lo que podría comprometer la confidencialidad, integridad y disponibilidad de la información empresarial.	Infraestructura de TI	Operativo	Falta de mantenimiento del sistema Capacidad insuficiente del servidor Configuración inadecuada	Cortes de energía Fallas de infraestructura Ataques cibernéticos	Interrupción del servicio ERP Paralización de operaciones Pérdidas económicas	2	3	6	Importante

ANÁLISIS DE RIESGOS												VALORACIÓN DEL RIESGO
Riesgo Inherente				Descripcion de los Controles Existentes				Riesgo Inherente				
Probabilidad (P)	Impacto (I)	P * I	Nivel del Riesgo (Inaceptable / Importante / Moderado / Tolerable / Aceptable)	Controles Existentes	Tipo de Control (Preventivo / Detectivo / Correctivo)	Frecuencia del Control (Varias Veces al Día / Diario / Semanal / Quincenal / Mensual / Trimestral / Semestral / Anual / Cuando se Requiere)	Automatización del Control (Manual / Semi Automático / Automático)	Probabilidad (P)	Impacto (I)	P * I	Nivel del Riesgo (Inaceptable / Importante / Moderado / Tolerable / Aceptable)	DECISIÓN (Tratar/No Tratar)
3	3	9	Inaceptable	No cuenta con ningún control establecido	Preventivo	Semanal	Manual	3	3	9	Inaceptable	Tratar
3	3	9	Inaceptable	No cuenta con ningún control establecido	Preventivo	Mensual	Manual	3	3	9	Inaceptable	Tratar
2	3	6	Importante	No cuenta con ningún control establecido	Preventivo	Semanal	Manual	2	3	6	Importante	Tratar
2	3	6	Importante	No cuenta actualmente	Preventivo	Trimestral	Manual	2	3	6	Importante	Tratar
2	3	6	Importante	No cuenta actualmente	Preventivo	Semanal	Manual	2	3	6	Importante	Tratar

Riesgo Inherente				VALORACIÓN DEL RIESGO	Tratamiento al Riesgo							Riesgo Residual (Esperado)			
Probabilidad (P)	Impacto (I)	P * I	Nivel del Riesgo (Inaceptable / Importante / Moderado / Tolerable / Aceptable)	DECISIÓN (Tratar/No Tratar)	TIPO DE RESPUESTA (Eliminar, Aceptar, Reducir, Transferir)	CONTROL NECESARIO	CONTROL Anexo A (ISO 27001)	RESPONSABLE	RECURSOS	PLAZO	PRIORIDAD	Probabilidad (P)	Impacto (I)	P * I	Nivel del Riesgo (Inaceptable / Importante / Moderado / Tolerable / Aceptable)
3	3	9	Inaceptable	Tratar	Reducir	Implementación de cifrado en tránsito (TLS), autenticación segura y gestión de accesos al sistema MFT	A.8.24 Uso de criptografía, A.5.15 Control de acceso	Seguridad TI / Administrador MFT	Certificados digitales, configuración TLS, políticas de acceso	1 mes	Alta	3	1	3	Moderado
3	3	9	Inaceptable	Tratar	Reducir	Implementación de autenticación multifactor (MFA), políticas de contraseñas robustas y bloqueo de intentos fallidos.	A.5.15 Control de acceso, A.5.17 Información de autenticación	Administrador de TI / Seguridad de la Información	Plataforma MFA, políticas de seguridad, capacitación a usuarios	1 mes	Alta	1	1	1	Aceptable
2	3	6	Importante	Tratar	Reducir	Validación automática de datos maestros, controles de integridad en la generación del archivo y revisión previa del pedido.	A.8.9 Gestión de la configuración, A.8.20 Integridad de la información	Analista Funcional ERP / Área de Compras	Reglas de validación en Odoo, documentación de procesos	1 mes	Alta	1	1	1	Aceptable
2	3	6	Importante	Tratar	Reducir	Implementación de controladores de dominio redundantes, monitoreo continuo y plan de continuidad del negocio (BCP).	A.5.30 Continuidad del negocio TIC, A.8.14 Redundancia	Área de Infraestructura TI	Servidores adicionales, herramientas de monitoreo, UPS	6 meses	Alta	2	2	4	Moderado
2	3	6	Importante	Tratar	Reducir	Implementación de autenticación multifactor (MFA), políticas de contraseñas seguras y gestión de accesos basada en roles (RBAC).	A.5.15 Control de acceso, A.5.17 Información de autenticación	Administrador de Sistemas / Seguridad de la Información	Plataforma MFA, políticas de seguridad, capacitación a usuarios	3 meses	Alta	2	2	4	Moderado

4.2 Diseñar el sistema de criptografía asimétrica



4.2.1 Especificación de Implementación de Encriptación

Considerando los requisitos establecidos en función de la necesidad expresada en el proyecto, se definen los requisitos técnicos del criptosistema para el servicio de e-mail y MFT. Adicionalmente, se implementará la encriptación y firma electrónica de los mensajes para asegurar la seguridad de los archivos. Del mismo modo, se debe garantizar el cumplimiento de los requisitos predefinidos que tienen como objetivo resolver los problemas identificados en el proyecto.

Para el desarrollo de la arquitectura de encriptado se utilizarán los siguientes criterios:

- Algoritmos de encriptado y firma
- Algoritmos de encriptado asimétrico
- Hash Function
- Certificado digital estandarizado

4.2.2 Definición de la Encriptación

En primer lugar, es importante determinar qué algoritmo se utilizará para cifrar y firmar digitalmente el e-mail, MFT y archivos en tránsito.

4.2.3 Algoritmo de Cifrado Asimétrico

Es necesario seguir las pautas indicadas en el NIST SP 800-57 para determinar qué algoritmo de cifrado usar. De esta forma, se verifica la selección de uno o más algoritmos de encriptación apropiados para la solución propuesta, siendo necesario excluir inicialmente aquellos cuyo uso no se recomienda actualmente. El primer paso es demostrar que las llaves de menos de 112 bits ya no se recomiendan. Así como tener en cuenta también que las llaves de 112 bits se permitirán hasta 2030, sin embargo, el tamaño de llave ya no se permitirá a partir de ese año. De igual manera, se puede observar que, si aceptará un tamaño de llave de 128 bits o más, por consiguiente, se realizará una búsqueda de un algoritmo con un tamaño de llave mayor o igual al nivel de seguridad de 128 bits, tomando en consideración una lista como punto de referencia.

Por otra parte, los algoritmos más destacados para el cifrado y firma son aquellos basados en RSA. Así mismo, el RSA considera una llave de 2048 bits en lugar de una clave de 3072 bits, esto es porque la potencia o capacidad de cómputo requerida para el cifrado y descifrado es muy elevado utilizando una llave de 3072 bits. Sin embargo, para acelerar el proceso de cifrado asimétrico, se recomienda utilizar RSA de 2048 bits. Por ello, se recomienda utilizar un rango de llaves de entre 256 a 383 bits, reduciendo ligeramente la velocidad del mecanismo de encriptación, pero aporta mayor nivel de seguridad a expensas del tamaño de su llave. Por lo tanto, el rango anterior que va desde 224 a 255 bits proporcionaba un proceso de cifrado más rápido, pero debido a que el tamaño de las llaves es más pequeño, pone limita el rango no válido de 160 a 223 bits.

Tabla 10
Tabla Comparativa de los Algoritmos de Cifrado AES, DES y RSA.

Tipo	Peso de archivo (KB)	Tiempo de cifrado (segundos)	Tiempo de descifrado (segundos)
AES	2048	0.56	0.81
DES		0.62	0.99
RSA		2.63	30.77
AES	4096	1.00	1.29
DES		1.23	1.56
RSA		4.50	64.31
AES	6144	1.65	1.97
DES		1.86	2.10
RSA		6.78	95.25
AES	8192	2.13	2.41
DES		4.47	2.76
RSA		9.03	126.59
AES	10240	2.81	3.09
DES		3.00	3.28
RSA		11.29	160.16

Para el rendimiento de los algoritmos de cifrado del cuadro en mención, los siguientes criterios son tomados en cuenta:

Duración aproximada

Se tiene en cuenta el tiempo que demora la ejecución. Es el tiempo que se necesita para generar un texto encriptado a partir de un texto sin formato.

Dimensión del búfer

El tamaño del búfer hace referencia a la variación en el uso de la memoria. Después de realizar un análisis, se llegó a la conclusión de que la solución requerirá el uso de algoritmos de cifrado asimétrico, y específicamente el RSA de 2048 bits. La especificación para el encriptado de los mensajes de e-mail indica que el algoritmo más adecuado para el cifrado asimétrico es también el RSA 2048 bits.

RSA es el mecanismo más utilizado en criptografía asimétrica y se utiliza para encriptar datos, archivos y firmas digitales. Su confiabilidad se basa en la longitud de las claves, que utilizan la factorización de números enteros. La información se presenta en forma de

números y se genera mediante la selección aleatoria y confidencial de dos números primos extensos. Como cualquier método asimétrico, se genera las llaves pública y privada para realizar el proceso de cifrado de la información, de esta manera solo el que posea la clave privada podrá descifrarlo.

4.2.4 Hash Funtion

Siguiendo las sugerencias de la NIST SP-800-57 Parte 3, se llevó a cabo un análisis para identificar los mecanismos hash que satisfacen los niveles de seguridad necesaria para realizar el cifrado utilizando RSA. El resultado del análisis indica que el hash recomendado para RSA-2048 es el SHA-256.

Estas operaciones poseen propiedades especiales que las hacen resistentes a los intentos de ataque dirigidos a comprometer los sistemas de información.

Para que una función hash sea considerada segura y criptográficamente robusta, debe ser:

- Unidireccionalidad: es difícil o imposible recuperar el mensaje original K a partir de su sumario H(K).
- Avalancha: cuando se realizan pequeños cambios en los valores, los hashes resultantes deberán ser significativamente distintos. Por ejemplo, si se altera un solo bit en un mensaje K, se espera que el sumario H(K) cambie al menos el 50% de sus bits.
- Resistencia a colisiones: ocurre cuando 2 mensajes diferentes, K y K', producen el mismo resumen, es decir, H(K) = H(K'). Dado que hay una cantidad infinita de textos distintos, es imposible que cada hash corresponda a un solo mensaje. Por lo tanto, es crucial que las funciones hash criptográficas tengan una probabilidad muy baja de generar colisiones.

En 2001, el NIST adoptó SHA-2 como estándar en lugar de SHA-1 para mejorar la seguridad. SHA-2, también creado por la NSA, es una versión avanzada de SHA-1 y está disponible en dos versiones diferentes:

- La primera produce un sumario de 256 bits y está creada para funcionar en procesamientos de 32 bits.
- La segunda produce un sumario de 512 bits y está creada para funcionar en procesamientos de 64 bits.

Basándonos en el análisis teórico, se recomienda utilizar una función hash con una fortaleza mínima de 128 bits. Se sugiere emplear el hash de 256 bits, ya que su procesamiento es mucho más rápido, lo que la hace adecuada para su uso en firmas y servicios de e-mail donde la velocidad de transacción es crucial, Además, se recomienda no aumentar el tiempo de latencia entre el envío y la recepción de mensajes, por lo que se sugiere utilizar SHA-256.

4.2.5 GNU Privacy Guard (GPG)

NU Privacy Guard constituye una herramienta orientada al cifrado y a la generación de firmas digitales, desarrollada por Werner Koch, la cual surge como una alternativa al sistema PGP. No obstante, su principal característica diferenciadora radica en que se trata de software libre, distribuido bajo la licencia GPL. Asimismo, GPG se encuentra alineado con el estándar RFC 4880 establecido por la IETF, conocido como OpenPGP, garantizando así la compatibilidad y seguridad en los procesos de protección de la información.

GnuPG permite encriptar y firmar sus datos y comunicaciones; cuenta con un versátil sistema de gestión de claves, junto con módulos de acceso a todo tipo de directorios de claves públicas. GnuPG, también conocido como GPG, cuenta con una línea de comandos con funciones para una fácil integración con otras aplicaciones. Hay disponible una gran cantidad de aplicaciones y bibliotecas frontend. GnuPG también brinda soporte para S/MIME y Secure Shell (ssh).

Desde su introducción en 1997, GnuPG es software libre (lo que significa que respeta su libertad). Se puede usar, modificar y distribuir libremente conforme a las disposiciones establecidas en la Licencia Pública General de GNU.

4.2.6 Algoritmos más utilizados de acuerdo con el estándar

El mecanismo de cifrado GPG debe seguir las recomendaciones del estándar PGP (IETF RFC 4880). En la sección 2 del estándar se especifican algoritmos adecuados para utilizar certificados PGP, asimismo posibilita el intercambio de mensajes tanto de salida como, de entrada.

Es importante seguir las recomendaciones del estándar PGP y utilizar los algoritmos que se sugiere, los cuales se consideran válidos para la solución. Se ha establecido que el conjunto ideal de algoritmos para el cifrado debe incluir RSA-2048 y SHA-256. El estándar también especifica que el algoritmo óptimo para el intercambio de llaves asimétricas para firmar debe

ser RSA, mientras que para el cifrado de los mensajes se debe utilizar AES-128 o una versión más avanzada.

4.2.7 Configuración de encriptación de las llaves utilizando GPG

En este punto se pretende establecer los criterios para la definición del formato que se utilizará para la creación de llaves sobre el servicio de e-mail y MFT, a fin de que estén claramente constituidos y normalizado dentro de la organización. Para ello, se seguirán las pautas recomendadas por el estándar GPG con relación a los algoritmos de cifrado, se deben considerar la longitud de las llaves y el tiempo durante el cual serán válidas. La estructura de la plantilla se ajustará a las especificaciones establecidas por dicho estándar, el cual proporciona las directrices necesarias para llevar a cabo la generación de llaves tanto para el cifrado como para la firma.

- Es necesario crear una plantilla para la generación de claves de cifrado que incluya información como el nombre, correo electrónico, identificación de la clave, fechas de creación y expiración, tipo y tamaño de la clave, nivel de seguridad, hash, verificación, encriptación, compresión, estatus, definición y uso que se le da a la clave.
- El asunto del mensaje de e-mail se utiliza para agregar una etiqueta que indique al algoritmo de cifrado que debe comenzar el proceso de encriptación. El algoritmo verificará si el destinatario es interno o externo. En el caso de ser interno, tratará de hallar la clave pública del destinatario o remitente en el servidor de claves. Si es externo, buscará un certificado X.509 compatible con PGP para cifrar dicho mensaje.
- Al configurar la longitud de las claves, se determina el tipo de algoritmo que se utilizará para cifrar la información, así como para del servicio de e-mail y del MFT.
- En relación con el ciclo de vida de las claves, es crucial considerar el lapso transcurrido desde su generación, publicación y distribución, el almacenamiento y la fecha de vencimiento. Si una clave caduca, se revocará y se generará automáticamente otra.

4.2.8 Configuración del Asunto

Es de suma importancia definir la configuración del campo "asunto" dentro de la estructura de cifrado, ya que determina si el mensaje debe ser encriptado antes de ser enviado al destinatario. De acuerdo con el valor de este campo, se establece una condición para el cifrado y se procede a hallar la clave pública del destinatario con el fin de encriptarla.

4.2.9 Configuración del Algoritmos y Tamaño de claves

Es fundamental en la configuración de las llaves establecer los algoritmos y longitudes de claves que se utilizarán y su nivel de seguridad correspondiente, a fin de garantizar la coherencia con el cifrado y la firma necesarios para los correos electrónicos que se envían a través del servicio G-Suite.

Al determinar qué algoritmo usar, se debe considerar el nivel de confidencialidad de la organización. Es recomendable usar algoritmos con una longitud mayor a 1024 bits. Los algoritmos por comparar en la sección de definición de algoritmos son el Estándar de Cifrado de Datos (DES), el Estándar de Cifrado Avanzado (AES) y el sistema de cifrado asimétrico de clave pública denominado RSA.

Se utilizarán los siguientes valores para el análisis comparativo:

- Rendimiento: calcula la estimación que demora el algoritmo en completar el proceso de cifrado.
- Longitud de clave: es importante para entender el cifrado los datos.
- Tipo de algoritmos: se cuenta con dos algoritmos, los cuales son simétrico y asimétrico, basados en el proceso y la clave.
- Conexión de encriptación: evalúa la cantidad de información a cifrar y debe ser considerado para disminuir la complejidad. Esta evaluación clasifica los niveles en tres categorías: baja, media y alta.
- Seguridad: la técnica de cifrado debe ser segura ante ataques de texto en claro o cifrado.
- Velocidad: evalúa la rapidez de los procesos criptográficos.
- Flexibilidad: la escalabilidad se refiere a la longitud de la clave y la diferencia del bloque.
- Uso de clave: se especifica si utilizan una clave diferente para el proceso criptográfico o si se usa la misma clave.
- Consumo de energía: se mide el consumo de energía durante el proceso de cifrado. Se distinguen dos grados: alto y bajo.

- Implementación: se considera si el componente físico y lógico son eficientes en la implementación de los algoritmos AES, DES y RSA.

Tabla 11
Valores de Referencia

Componente	DES	AES	RSA
Creación	1977	2000	1978
Tamaño	56 bits	128, 192,256 bits	Superior a 1024 Bits
Categoría	Simétrica	Simétrica	Asimétrica
Vinculación de Encriptación	Bajo	Bajo	Alto
Seguridad	No adecuado	Completamente seguro	Ataque de tiempo
Velocidad de Simulación	Rápido	Rápido	Lento
Flexibilidad	Escalable	No escalable	No escalable
Llave empleada	Solo una para el encriptado y desencriptado	Solo una para el encriptado y desencriptado	Dos llaves una para encriptar y otra para desencriptar
Consumo de Energía	Bajo	Bajo	Bajo
Implementación Hardware y software	Mejor en hardware que en software	Más rápido y eficiente	Alto costo computacional
Numero de rondas	16	10/12/2014	N/A
Utilidad	Confidencialidad	autenticidad de origen	no repudio
Funcionalidad	Cifrado de mensajes, cifrado de archivos	Cifrado de mensajes, cifrado de archivos, cifrado de contraseñas	Cifrado de mensajes, firma digital, intercambio de claves

Se puede observar que AES es el algoritmo más moderno en cuanto a su fecha de creación, mientras que RSA contiene mayor que DES y AES, siendo DES el que tiene la longitud de clave más pequeña, por lo que se considera menos seguro que RSA y AES. Son más rápidos y consumen menos energía en los procesos criptográfico; sin embargo, también son más vulnerables que los algoritmos asimétricos, que requieren dos claves para cifrar y descifrar.

4.2.10 Modelo de Arquitectura del Criptosistema Asimétrico SMTP y MFT

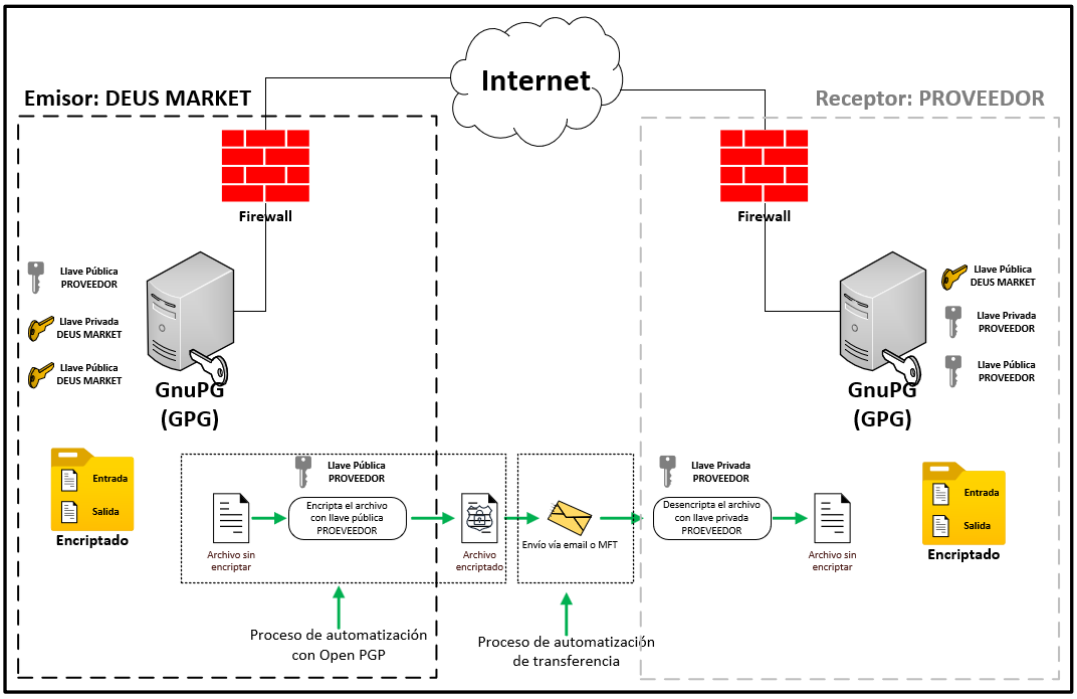
En esta sección o etapa, se detalla cual es la arquitectura y sus componentes que considerar. Inicialmente se observa del lado del Emisor (izquierda de la figura 11), que se cuenta con el repositorio a encriptar, así como el proceso de encriptación (considerando las llaves Privada y Pública) que va desde seleccionar el archivo plano (sin encriptar) que luego pasa por el

proceso de encriptación para luego ser enviado a través del proceso automatizado de transferencia.

Por el lado del Receptor (derecha de la figura 11) se observa que recibe el archivo encriptado para luego realizar el proceso de desencriptación aplicado las llaves generadas por el Receptor y recibidas por el lado del Emisor.

Dando como resultado que la transferencia de archivos sea de forma segura aplicando la encriptación por GPG (GNU Privacy Guard) entre el Emisor y Receptor.

Figura 36
Arquitectura criptográfica asimétrica aplicada a SMTP y MFT



Nota: Modelo de Arquitectura, por elaboración propia, 2026

4.2.11 Componentes de Arquitectura del Criptosistema

4.2.11.1 Servidor GPG

Dado a la importancia mundial del sistema de cifrado, varias empresas requieren llevar a cabo un software que funcione con GPG. En el año de 1997, PGP Inc. creo un estándar llamado PGP al Internet Engineering Task Force (IETF), permitiendo que usara este nombre, que describe tanto el nuevo estándar como todas las aplicaciones que puedan utilizar este tipo de criptografía. El IETF aceptó esta propuesta y lanzó PGP. Para este proyecto, utilizaremos GNU Privacy Guard (GPG) de uso y licenciado de forma Libre (General Public

License - GPL) basado en el estándar PGP para cifrar la información en reposo, en tránsito mediante el servicio de e-mail (correo electrónico).

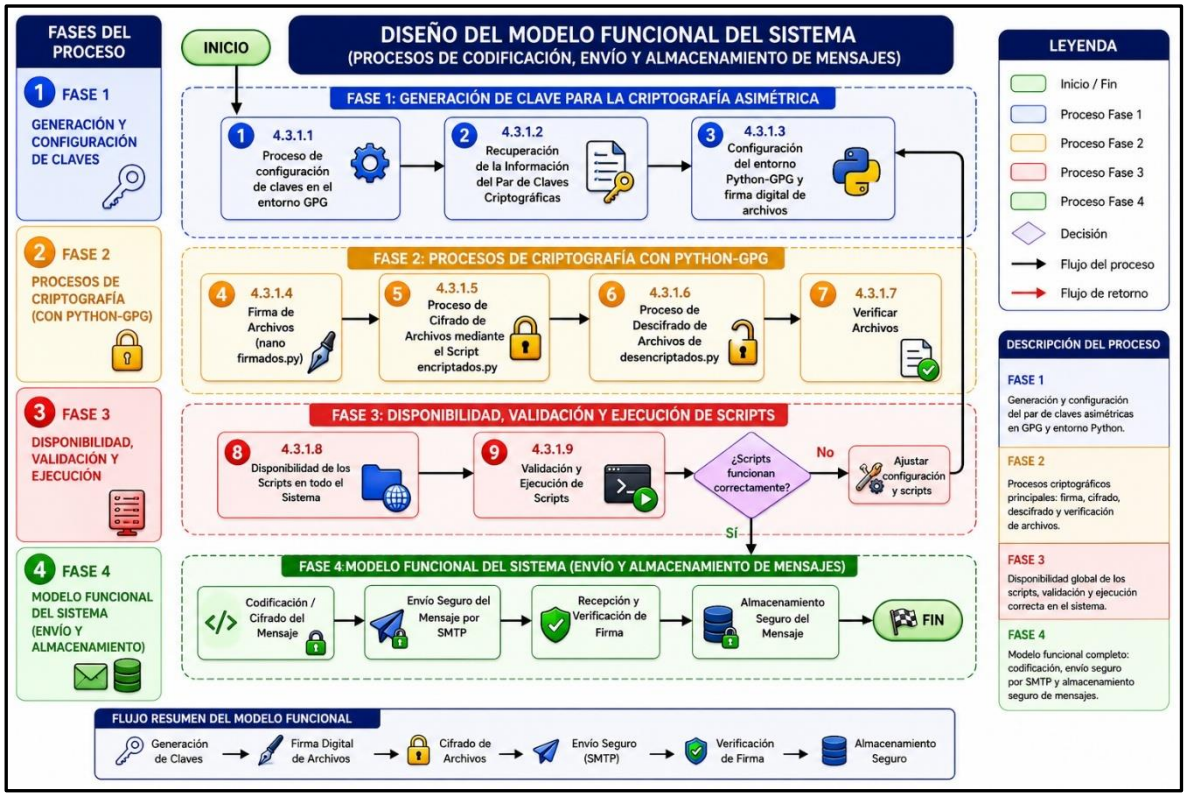
4.2.11.2 Cambio de Llaves

Estas claves se generarán a través del GPG de DEUSMARKET una clave privada y clave pública, esta última se distribuirá a través de un servidor en la nube de tipo público. Así mismo, se remitirá al PROVEEDOR para el cifrado de los ficheros transferidos, la llave privada es configurada en servidor diferente para protegerlo de ataques o afectación, utilizándolo también para descifrar archivos.

4.2.11.3 Firewall

Componente de seguridad de red que se utiliza para controlar el acceso a una red privada o a un sistema informático individual. Actuando como una barrera que bloquea el tráfico no autorizado desde o hacia la red o sistema protegido evitando el acceso no autorizado. Este equipo, se coloca entre la red privada o el sistema protegido y la red externa, que generalmente es Internet, examinando cada paquete de datos que intenta pasar a través de él aplicando las reglas de seguridad configuradas. El Firewall es una parte fundamental del proyecto porque permitirá conexiones desde el remitente (DEUSMARKET) hacia el destinatario (PROVEEDOR) y viceversa.

4.3 Diseñar del sistema funcional para la codificación



4.3.1 Proceso de generación de clave para la criptografía de tipo asimétrica

4.3.1.1 Proceso de configuración de claves en el entorno GPG

Para llevar a cabo la configuración de claves GPG, se empleará el paquete GnuPG, una herramienta que viene incorporada de manera predeterminada en diversas distribuciones del sistema operativo Linux, entre ellas Ubuntu y CentOS. Este software está orientado a reforzar la seguridad de las comunicaciones electrónicas, ya que permite implementar mecanismos de protección más robustos y avanzados.

Se empleará el siguiente comando para actualizar el índice de los repositorios locales e instalar los paquetes GPG y GnuPG en el sistema:

```
1. sudo apt-get update
2. sudo apt-get install gnupg
```

Para iniciar el uso de GPG en el proceso de cifrado, es necesario generar un par de claves criptográficas. Para ello, se ejecutará el siguiente comando:

```
1. gpg --gen-key
```

```
1. Real name: Deusmarket
2. Email address: ventasonline@deusmarket.pe
3. You selected this USER-ID:
4.     "Deusmarket <ventasonline@deusmarket.pe>"
```

4.3.1.2 Recuperación de la Información del Par de Claves Criptográficas

Para visualizar las claves generadas, se utilizará el siguiente comando, el cual permitirá mostrar el listado completo de llaves criptográficas creadas en el sistema. De esta manera, será posible consultar y verificar todas las claves generadas previamente.

```
1. gpg --list-key
```

```
1. pub   rsa3072 2023-05-07 [SC] [expires: 2025-05-06]
2.       CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1
3. uid           [ultimate] Deusmarket
<ventasonline@deusmarket.pe>
```

4.3.1.3 Configuración del entorno Python-GPG y firma digital de archivos

Una vez generadas las claves criptográficas, se procederá con la instalación del módulo python-gnupg, el cual forma parte de GnuPG y permite establecer la comunicación entre GnuPG y el lenguaje Python 3. Mediante este módulo, será posible desarrollar scripts en Python orientados a la gestión de procesos criptográficos, tales como:

- Generación de firmas digitales para archivos.
- Cifrado de información.
- Descifrado de archivos cifrados.
- Verificación de firmas digitales.

Asimismo, se instalará el paquete fs, el cual facilitará la apertura, lectura y escritura de archivos de prueba dentro del sistema. Para iniciar este proceso, se actualizará el índice de paquetes y posteriormente se realizará la instalación de ambos módulos mediante el gestor pip.

```
1. sudo apt-get update
2. sudo apt-get install python3-pip
3. sudo apt-get install python3-gnupg
```

Con estos paquetes en su lugar, podemos pasar a crear los scripts y los archivos de prueba.

Para almacenar los scripts y los archivos de prueba, crearemos un directorio llamado python-prueba:

```
1. mkdir python-prueba
```

Mover a este directorio:

```
1. cd python-prueba
```

Se procederá a la creación de tres archivos de prueba denominados test1.txt, test2.py y test3.txt, los cuales serán utilizados para realizar las pruebas correspondientes del sistema de cifrado y verificación.

```
1. echo " Este es el primer archivo de prueba" > test1.txt
2. echo "print(' Este archivo de prueba es una secuencia de
comandos de Python ')" > test2.py
3. echo " Este es el último archivo de prueba" > test3.txt
```

Con el propósito de generar firmas digitales independientes para los archivos de prueba, se desarrollará un script denominado firmados.py, el cual tendrá como función procesar todos los archivos ubicados en el directorio desde donde sea ejecutado. La firma digital actuará como un mecanismo de validación que permitirá validar la autenticidad y garantizar la integridad de los documentos, además de registrar una referencia temporal asociada a su creación.

Las firmas digitales generadas se almacenarán en un directorio denominado firmas/, el cual será creado automáticamente durante la ejecución del script.
Abriremos un fichero y lo llamaremos firmados.py con el nano:

```
1. nano firmados.py
```

Se importarán las librerías necesarias para la ejecución del código, incluyendo los módulos fs y os, los cuales permiten gestionar y manipular archivos dentro del sistema. Asimismo, se incorporará el paquete gnupg, encargado de implementar las funcionalidades criptográficas

requeridas en el desarrollo del proceso.

```
1. #!/usr/bin/env python3
2. import os
3. import fs
4. from fs import open_fs
5. import gnupg
```

A continuación, se configurará el directorio en el cual GnuPG localizará las claves de cifrado. De manera predeterminada, GnuPG almacena dichas claves en el directorio. gnupg; por ello, se establecerá esta ruta utilizando el nombre de usuario correspondiente del sistema.

```
1. gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
```

A continuación, se define la variable home_fs, la cual contendrá la ruta del archivo en ejecución como un objeto de tipo archivo, permitiendo al script trabajar correctamente dentro del directorio actual.

```
1. home_fs = open_fs(".")
```

Una vez completada la implementación del código, la estructura final deberá visualizarse de la siguiente manera:

```
1.  #!/usr/bin/env python3
2.
3.  import os
4.  import fs
5.  from fs import open_fs
6.  import gnupg
7.
8.  gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
9.  home_fs = open_fs(".")
```

Posteriormente, se añadirá el código encargado de verificar la existencia del directorio denominado firmas/; en caso de no encontrarse creado, este será generado automáticamente por el sistema.

```
1. if os.path.exists("firmas/"):
2.         print("El directorio Firmas ya está creado")
3. else:
4.         home_fs.makedir(u"firmas")
5.         print("Se creo el directorio firmas")
```

Se implementará una estructura iterativa for encargada de almacenar los archivos identificados. Posteriormente, se recorrerá el directorio actual con el fin de localizar los

ficheros existentes y agregar un identificador correspondiente a cada uno dentro de la matriz files_dir.

```
1. files_dir = []
2.
3. files = [f for f in os.listdir(".") if os.path.isfile(f)]
4. for f in files:
5.     files_dir.append(f)
```

Posteriormente, el script procederá a generar firmas digitales individuales para cada uno de los archivos contenidos en la matriz files_dir. Para permitir el acceso a la clave privada durante este proceso, dicha clave será desbloqueada mediante la contraseña previamente configurada “Pr0yect0PP2”.

```
1. for x in files_dir:
2.     with open(x, "rb") as f:
3.         stream =
gpg.sign_file(f,passphrase="Pr0yect0PP2",detach = True,
output=files_dir[files_dir.index(x)]+".sig")
4.         os.rename(files_dir[files_dir.index(x)]+".sig",
"firmas/"+files_dir[files_dir.index(x)]+".sig")
5.         print(x+" ", stream.status)
```

Cuando termine, las firmas trasladaran a la carpeta firmas. El script final se vera de la siguiente manera:

4.3.1.4 Firma de Archivos (nano firmados.py)

```
1.  [label ~/python-test/firmados.py]
2.  #!/usr/bin/env python3
3.
4.  import os
5.  import fs
6.  from fs import open_fs
7.  import gnupg
8.
9.   gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
10. home_fs = open_fs(".")
11.
12. if os.path.exists("firmas/"):
13.     print("El directorio Firmas ya está creado ")
14. else:
15.     home_fs.makedir(u"firmas")
16.     print("Se creo el directorio firmas ")
17.
18. files_dir = []
19.
20. files = [f for f in os.listdir(".") if os.path.isfile(f)]
21. for f in files:
22.     files_dir.append(f)
23.
24. for x in files_dir:
25.     with open(x, "rb") as f:
26.         stream =
gpg.sign_file(f,passphrase="Pr0yect0PP2",detach = True,
output=files_dir[files_dir.index(x)]+".sig")
27.         os.rename(files_dir[files_dir.index(x)]+".sig",
"firmas/"+files_dir[files_dir.index(x)]+".sig")
28.         print(x+" ", stream.status)
```

4.3.1.5 Proceso de Cifrado de Archivos mediante el Script encriptados.py

Al ejecutar el código de cifrado almacenado en una carpeta del sistema, los archivos de dicha ubicación son copiados y posteriormente encriptados en la carpeta encriptados/. Para llevar a cabo el proceso de cifrado, se utilizará la clave pública asociada al correo electrónico configurado previamente: ventasonline@deusmarket.pe.

Se procederá a crear un nuevo archivo denominado encriptados.py, el cual contendrá el código correspondiente al proceso de cifrado de archivos.

```
1. nano encriptados.py
```

En primer lugar, se importarán las librerías necesarias para configurar el directorio de inicio de GnuPG, así como para crear la variable correspondiente al directorio actual de trabajo.

```
1.  #!/usr/bin/env python3
2.
3.  import os
4.  import fs
5.  from fs import open_fs
6.  import gnupg
7.
8.  gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
9.  home_fs = open_fs(".")
```

A continuación, se incorporará el código encargado de verificar si en el directorio actual existe una carpeta denominada encriptados/; en caso de no encontrarse, esta será creada automáticamente por el sistema.

```
1. if os.path.exists("encriptados/"):
2.         print("El directorio encriptados ya está creado ")
3. else:
4.         home_fs.makedir(u"encriptados")
5.         print("Se creo el directorio encriptados ")
```

Previo al inicio del proceso de búsqueda de archivos para su cifrado, se definirá una estructura tipo matriz destinada a almacenar los ficheros identificados durante la ejecución del script.

```
1. files_dir = []
```

A continuación, se implementará un bucle for encargado de recorrer el directorio de trabajo, con el objetivo de localizar los archivos disponibles y añadirlos a la matriz previamente creada para su posterior procesamiento.

```
1. files = [f for f in os.listdir(".") if os.path.isfile(f)]
2. for f in files:
3.     files_dir.append(f)
```

Finalmente, se implementará un ciclo que permitirá cifrar todos los archivos ubicados en el directorio de trabajo. Una vez concluido el proceso, los archivos cifrados serán transferidos de manera automática al directorio encriptados/. Para el cifrado se empleará la clave pública asociada al identificador de correo electrónico ventasonline@deusmarket.pe, el cual será

utilizado como referencia para localizar la clave correspondiente durante la operación de encriptación.

```
1. for x in files_dir:
2.     with open(x, "rb") as f:
3.         status =
gpg.encrypt_file(f,recipients=["ventasonline@deusmarket.pe"],outpu
t= files_dir[files_dir.index(x)]+".gpg")
4.         print("ok: ", status.ok)
5.         print("status: ", status.status)
6.         print("stderr: ", status.stderr)
7.         os.rename(files_dir[files_dir.index(x)] + ".gpg",
'encriptados/' +files_dir[files_dir.index(x)] + ".gpg")
```

El script encriptados.py una vez configurado se vera de la siguiente manera:

```
 1. import os
 2. import fs
 3. from fs import open_fs
 4. import gnupg
 5. gpg = gnupg.GPG (gnupghome ="/home/deusmarket/. gnupg ")
 6. home_fs = open_fs(".")
 7. if os.path.exists("encriptados/"): print("Existe un directorio
encriptados")
 8. else:
 9. home_fs.makedir(u"encriptados")
10. print("Directorio encriptados creado ")
11. archivos_dir = []
12. archivos = [f for f in os.listdir(".") if os.path.isfile(f)]
13. for f in archivos:
14. archivos_dir.append(f)
15. for x in files_dir:
16. with open(x, "rb") as f:
17. status=
gpg.encrypt_file(f,recipients=["ventasonline@deusmarket.pe"],outpu
t= archivos_dir[archivos_dir.index(x)]+".gpg")
18. print("ok: ", estado.ok)
19. print("estado: ", estado.status)
20. print("stderr: ", estado.stderr)
21. os.rename(archivos_dir[archivos_dir.in dex(x)]
+".gpg","encriptados/"+ archivos_dir[archivos_dir.index(x)]+
".GPG")
```

4.3.1.6 Proceso de Descifrado de Archivos de desencriptados.py

El script de descifrado presenta una lógica similar al de cifrado; sin embargo, está diseñado para ejecutarse dentro del directorio encriptados/. Al iniciar la ejecución de desencriptar.py, el sistema procederá a identificar las claves criptográficas registradas en el sistema. Al

término del proceso de descifrado, los archivos resultantes serán guardados automáticamente en un nuevo directorio denominado desencriptados/.

Se procederá a crear un nuevo archivo denominado desencriptados.py utilizando el editor de texto nano, el cual contendrá el script encargado del proceso de descifrado de archivos.

```
1. nano desencriptados.py
```

A continuación, se incorporarán los módulos necesarios para dar inicio al desarrollo del código, los cuales permitirán implementar las funcionalidades requeridas para el proceso de descifrado de archivos.

```
1.  #!/usr/bin/env python3
2.
3.  import os
4.  import fs
5.  from fs import open_fs
6.  import gnupg
7.
8.  gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
9.  home_fs = open_fs(".")
```

A continuación, se definirán dos matrices que serán utilizadas para el almacenamiento de datos durante la ejecución del script, permitiendo organizar y gestionar la información procesada en el proceso de descifrado.

```
1. files_dir = []
2. files_dir_clean = []
```

El propósito de esta etapa es garantizar que los archivos descifrados se almacenen en un directorio independiente, evitando así la mezcla con los archivos cifrados y facilitando su posterior localización. Para ello, se implementará un mecanismo que verifique la existencia del directorio desencriptados/ dentro de la ruta actual; en caso de no existir, este será creado automáticamente por el sistema.

```
1. if os.path.exists("desencriptados/"):
2.     print("El directorio desencriptado ya está creado ")
3. else:
4.     home_fs.makedir(u"desencriptados/")
5.     print("Se creo el directorio desencriptado ")
```

Se procederá a examinar el directorio con el fin de incorporar los nombres de los archivos en la matriz files_dir

```
1. files = [f for f in os.listdir(".") if os.path.isfile(f)]
2. for f in files:
3.     files_dir.append(f)
```

Todos los archivos cifrados incluyen la extensión .gpg en su nombre, lo cual permite identificarlos como datos encriptados. No obstante, al momento de realizar el proceso de descifrado, estos archivos serán almacenados sin dicha extensión, dado que ya no se encuentran cifrados, permitiendo así una mejor organización y claridad en la gestión de los archivos resultantes.

Para hacer esto, recorra los files_dir matriz y elimine la .gpg extensión de cada nombre de archivo:

```
1.     for x in files_dir:
2.             length = len(x)
3.             endLoc = length - 4
4.             clean_file = x[0:endLoc]
5.             files_dir_clean.append(clean_file)
```

Los nuevos nombres de archivo "limpiados" se almacenan dentro de la file_dir_clean matriz.

A continuación, se realizará el recorrido de los archivos con el objetivo de proceder a su descifrado. Durante este proceso, se ingresará la clave de acceso correspondiente para desbloquear la clave privada, permitiendo así completar la operación de descifrado de manera segura.

```
1. for x in files_dir:
2.     with open(x, "rb") as f:
3.         status = gpg.decrypt_file(f,
passphrase="Pr0yect0PP2",output=files_dir_clean[files_dir.index(x)
])
4.         print("ok: ", status.ok)
5.         print("status: ", status.status)
6.         print("stderr: ", status.stderr)
7.         os.rename(files_dir_clean[files_dir.index(x)],
"desencriptados/" + files_dir_clean[files_dir.index(x)])
```

Su archivo de script se verá así cuando haya terminado:

```
#!/usr/bin/env python3

import os
import fs
from fs import open_fs
import gnupg

gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
home_fs = open_fs(".")

files_dir = []
files_dir_clean = []

if os.path.exists("desencriptados/"):
    print("El directorio desencriptado ya existe")
else:
    home_fs.makedir(u"desencriptados/")
    print("Se creo el directorio desencritados")

files = [f for f in os.listdir(".") if os.path.isfile(f)]
for f in files:
    files_dir.append(f)

for x in files_dir:
    length = len(x)
    endLoc = length - 4
    clean_file = x[0:endLoc]
    files_dir_clean.append(clean_file)

for x in files_dir:
  with open(x, "rb") as f:
     status = gpg.decrypt_file(f, passphrase="
Pr0yect0PP2",output=files_dir_clean[files_dir.index(x)])
        print("ok: ", status.ok)
        print("status: ", status.status)
        print("stderr: ", status.stderr)
        os.rename(files_dir_clean[files_dir.index(x)],
"desencriptados/" + files_dir_clean[files_dir.index(x)])
```

4.3.1.7 Verificar Archivos

El script final tiene como función validar previamente otros scripts antes de proceder con su ejecución. En este contexto, su comportamiento guarda similitud con el proceso de verificación de archivos; sin embargo, incorpora la capacidad adicional de ejecutar aquellos scripts que han sido correctamente validados. Su funcionamiento consiste en recibir como parámetro el nombre de un script y, posteriormente, comprobar la validez de su firma digital. En caso de que la verificación resulte satisfactoria, se genera un mensaje en la consola y se

ejecuta el archivo correspondiente. Por el contrario, si la validación no es exitosa, se muestra un mensaje de error y se interrumpe la ejecución del script.

Se procederá a crear un nuevo archivo denominado verificar_archivo.py, el cual estará destinado a la implementación de las funciones necesarias para la verificación de archivos dentro del sistema.

```
1. nano verificar_archivo.py
```

En primer lugar, se importarán las bibliotecas requeridas para el desarrollo del script, así como la configuración de los directorios de trabajo necesarios.

```
1.  #!/usr/bin/env python3
2.
3.  import os
4.  import fs
5.  from fs import open_fs
6.  import gnupg
7.
8.  gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
9.  home_fs = open_fs(".")
```

Para que el script pueda ejecutarse correctamente, es necesario almacenar el nombre del archivo que será verificado y ejecutado. Para ello, se definirá una nueva variable denominada script_to_run, la cual contendrá la referencia al archivo correspondiente.

```
1. script_to_run = str(sys.argv[1])
```

La variable toma como entrada el primer argumento proporcionado y lo almacena en la variable previamente definida. Seguidamente, el script procederá a abrir el archivo de firma correspondiente, realizar la verificación del archivo script_to_run mediante dicha firma y, ejecutar el archivo.

```
1.  with open("../../firmas/" + script_to_run + ".sig", "rb") as
f:
2.        verify = gpg.verify_file(f, script_to_run)
3.        print(script_to_run + " ", verify.status)
4.        if verify.status == "firma valida":
5.             print("Firma valida, ejecutando codigo...")
6.             exec(open(script_to_run).read())
7.        else:
8.              print("Firma invalidad, ")
9.              print("Saliendo del codigo de ejecucion")
```

El script terminado se verá así:

```
1.  #!/usr/bin/env python3
2.
3.  import os
4.  import sys
5.  import fs
6.  from fs import open_fs
7.  import gnupg
8.
9.  gpg = gnupg.GPG(gnupghome="/home/deusmarket/.gnupg")
10. home_fs = open_fs(".")
11.
12. script_to_run = str(sys.argv[1])
13.
14. with open("../../firmas/" + script_to_run + ".sig", "rb") as
f:
15.      verify = gpg.verify_file(f, script_to_run)
16.      print(script_to_run + " ", verify.status)
17.      if verify.status == "firmas valida":
18.          print("Firma valida, ejecutando codigo...")
19.          exec(open(script_to_run).read())
20.      else:
21.          print("Firma invalidad, ")
22.          print("Saliendo del codigo de ejecucion ")
```

Se ha concluido la creación de los scripts; sin embargo, en su estado actual únicamente pueden ser ejecutados desde el directorio actual. Por ello, en el siguiente paso se procederá a modificar sus permisos con el fin de habilitar su ejecución de manera global en el sistema.

4.3.1.8 Disponibilidad de los Scripts en todo el Sistema

Para facilitar su uso, hagamos que los scripts sean ejecutables desde cualquier directorio o carpeta del sistema y lo colocaremos dentro de nuestro archivo $PATH. Usaremos el chmod comando para otorgar permisos ejecutables al propietario de los archivos, su usuario no root:

```
1. chmod +x *.py
```

Ahora para encontrar su $PATH configuración, ejecute el siguiente comando:

```
1. echo $PATH
```

```
1. Output
2. -bash:
/home/local/bin:/home/local/.local/bin:/usr/local/sbin:/usr/local/
bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/sna
p/bin
```

Los archivos almacenados en el $PATH son accesibles desde cualquier carpeta dentro del sistema. Se colocarán las secuencias de comandos en cualquier lugar dentro del $PATH, ahora moveremos las secuencias de comandos del python-prueba a /usr/local/bin/.

```
1. sudo mv encriptados.py /usr/local/bin/encriptados
2. sudo mv desencriptados.py /usr/local/bin/desencriptados
3. sudo mv firmadossigndetach.py /usr/local/bin/firmados
4. sudo mv verificar_archivo.py /usr/local/bin/verificar_archivo
5. sudo mv verificar.py /usr/local/bin/verificar
```

Ahora las secuencias de comandos se pueden ejecutar en cualquier parte del sistema simplemente ejecutando el nombre de la secuencia de comandos junto con los argumentos que la secuencia de comandos pueda tomar de la línea de comandos. En el siguiente paso, veremos algunos ejemplos de cómo usar estos scripts.

4.3.1.9 Validación y Ejecución de Scripts

Ahora que los scripts han sido añadidos a la variable del entorno $PATH, es posible ejecutarlos desde cualquier directorio dentro del servidor, sin necesidad de ubicarlos en la carpeta donde fueron creados originalmente.

En primer lugar, se debe verificar si aún se encuentra ubicado dentro del directorio python-prueba, utilizando el comando pwd, el cual permite visualizar la ruta actual de trabajo en el sistema.

```
1. pwd
```

La salida esperada debe ser:

```
1. Output
2. /home/deusmarket/python-test
```

Previamente, se crearon tres archivos de prueba durante el desarrollo del tutorial. A continuación, se debe ejecutar el comando ls -l con el fin de mostrar el listado detallado de los archivos almacenados en el directorio actual.

```
1. ls -l
```

Debería ver tres archivos almacenados en su python-test carpeta:

```
1. Output
2. -rw-rw-r-- 1 proyecto proyecto 07 May 15 10:08 test1.txt
3. -rwxrwxr-x 1 proyecto proyecto 07 May 15 10:08 test2.py
4. -rw-rw-r-- 1 proyecto proyecto 07 May 15 10:08 test3.txt
```

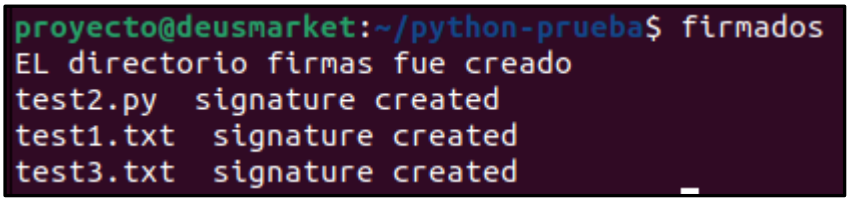
Se procederá a realizar las pruebas de los scripts sobre los tres archivos previamente creados. Antes de aplicar el proceso de cifrado, es posible visualizar de forma rápida el contenido de cada archivo utilizando el comando cat, tal como se muestra a continuación:

```
1. cat test1.txt
```

```
1. Output
2. Este es el primer archivo de prueba
```

Se iniciará el proceso de creación de firmas digitales independientes para cada uno de los archivos. Para ello, se ejecutará el script firmado desde el directorio actual, lo que permitirá generar las firmas correspondientes de manera individual para cada archivo.

```
1. firmados
```



En el resultado obtenido se observa que el script identificó que el directorio firmas/ no existía, por lo que procedió a crearlo automáticamente. Posteriormente, se generó correctamente las firmas digitales correspondientes a los archivos.

```
1. ls -l
```

```
proyecto@deusmarket:~/python-prueba$ ls -l
total 16
drwxrwxr-x 2 proyecto proyecto 4096 Keg   7 17:22 firmas
-rw-rw-r-- 1 proyecto proyecto   37 Keg   7 16:25 test1.txt
-rwxrwxr-x 1 proyecto proyecto   73 Keg   7 16:25 test2.py
-rw-rw-r-- 1 proyecto proyecto   38 Keg   7 16:25 test3.txt
```

Se puede observar el nuevo directorio firmas dentro del listado de elementos del sistema, el contenido de dicha carpeta y las firmas generadas.

Para listar todas las firmas disponibles, se debe ejecutar el siguiente comando:

```
1. ls -l firmas/
```

```
proyecto@deusmarket:~/python-prueba$ ls -l firmas
total 12
-rw-rw-r-- 1 proyecto proyecto 659 Keg   7 17:22 test1.txt.sig
-rw-rw-r-- 1 proyecto proyecto 659 Keg   7 17:22 test2.py.sig
-rw-rw-r-- 1 proyecto proyecto 659 Keg   7 17:22 test3.txt.sig
```

Las firmas almacenadas como archivos separados pueden identificarse mediante la extensión sig. Asimismo, es posible utilizar el comando cat para visualizar su contenido, como en el caso del archivo test1.txt.sig

```
1. cat firmas/test1.txt.sig
```

```
proyecto@deusmarket:~/python-prueba$ cat firmas/test1.txt.sig
-----BEGIN PGP SIGNATURE-----

iQGzBAABCgAdFiEEzRWh5ve/4qtvMQsP4LVshuBTlMEFAmRYJKgACgkQ4LVshuBT
lMGyjgv+P0xKD/lvPrSCaRUmhSmfgnQzop0PTeJ6/DIIyiLFZM9OB262dAm1OQGs
fTSDWl4eCVXq/KYg2VBPNccuP0NcqndtrpIz/QMMZWE2yARiYFI9IFqiPd+YIVuY
94OKL4cUJa8mH4SUmdafCOGF8yahHTkX/jS8I18oRTC1yptkxE3PkOZgyCdMTeWi
HAgoUMOUwntU3yrRhX20olpJBICqaj+U4PI+QsZ+l5xL9YmkTRss1KiHZaSN+ZHv
YB+lVfDP/APfoocmVEuCVMJ4slqRBh+1wzvYViaS9WPr4xX/MkVWFB84gY3tnYWJ
HwXmI7EsqKsDzrbZoJKAM3+ZrDrwvyq9ZxIqbUn1G66IU+GmQXHp6K97UxYCHhyh
to1IspXEeJP/SfnWcY4Dvt2rDfnV5OdUJH7zCO5TKsT7xElqod/WsYxOVTEUk9k0
GfzgZ7lKcDgPIXiiWnxDN+As2nMfAjPU33r3XD3ysvbRV3nxaqX7g6fmSSfl3iTZ
cVdiwlwD
=wk8F
-----END PGP SIGNATURE-----
```

El resultado mostrado corresponde a la firma independiente generada a partir del archivo test1.txt.Una vez que las firmas han sido creadas y almacenadas correctamente, se puede

continuar con el proceso de cifrado de los archivos. Para ello, se debe ejecutar el script de encriptación correspondiente.

```
proyecto@deusmarket:~/python-prueba$ encriptados
Se creo el directorio encriptados
ok:  True
status:  encryption ok
stderr:  [GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] ENCRYPTION_COMPLIANCE_MODE 23
[GNUPG:] BEGIN_ENCRYPTION 2 9
[GNUPG:] END_ENCRYPTION

ok:  True
status:  encryption ok
stderr:  [GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] ENCRYPTION_COMPLIANCE_MODE 23
[GNUPG:] BEGIN_ENCRYPTION 2 9
[GNUPG:] END_ENCRYPTION

ok:  True
status:  encryption ok
stderr:  [GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] ENCRYPTION_COMPLIANCE_MODE 23
[GNUPG:] BEGIN_ENCRYPTION 2 9
[GNUPG:] END_ENCRYPTION
```

En la salida generada se puede apreciar que el script creó automáticamente el directorio encriptados. Asimismo, se verifica que todos los archivos fueron cifrados de manera correcta.

A continuación, se ejecuta nuevamente el comando ls -l para listar el contenido del directorio y confirmar la presencia de los archivos cifrados.

```
1. ls -l
```

```
proyecto@deusmarket:~/python-prueba$ ls -l
total 20
drwxrwxr-x 2 proyecto proyecto 4096 Keg   7 17:30 encriptados
drwxrwxr-x 2 proyecto proyecto 4096 Keg   7 17:22 firmas
-rw-rw-r-- 1 proyecto proyecto   37 Keg   7 16:25 test1.txt
-rwxrwxr-x 1 proyecto proyecto   73 Keg   7 16:25 test2.py
-rw-rw-r-- 1 proyecto proyecto   38 Keg   7 16:25 test3.txt
```

A continuación, se analizará el archivo test1.txt para observar cómo se presenta el mensaje tras haber sido cifrado.

```
1. cat desencriptados/test1.txt.gpg
```

```
proyecto@deusmarket:~/python-prueba$ cat encriptados/test1.txt.gpg
-----BEGIN PGP MESSAGE-----

hQGMAzqeO1LuiEtcAQv+JsCJrQT5cY4i1bQ17oqYQLXCih0SVElqv//phdFw5gNp
iaVnw75r1PeICzatfh69l04hvLn6rMnt9JEIU62hlTIvpx677hxP/D2xh+8/v7sx
U3MKqa7naOmlzxmtYz+5H8cmxLXJ1PgaA6D57vYUEzIKdiHKBaKS3DAfW+vpgXJ+
pxTjddpatYaJqo4cyTYiYkE+QxLRSKuOMY5gWQjWKf3+yvQlT5xx+cJre/aCgoU5
yHq1SxROWVGPuZxpVLJ00gARvHWEZZdFuiLCdtYFCznjqE0oyxnnAvTyapdwcZME
3C3MAavanPC+4Om5zK3wYLLPR+Ja+Sd+s/bhPM+vQoFts0gsPMTlqUUoOgySJIU8
9riXQqKdUh3brwVKngmUpLaB51Irl87FG+xNvN4K5iq6PyyzeKVrAzgoaT4A+dBf
fYGYla6WXWtjr1U4e5YIdEN1rSJGe/b3Liwk3+ubtKX3l+xYE14P8A3pn4zjt4uP
UdpOENbv5I23U5J1FGUI0l8B29Wa4InprqTz1u+5Iq6K1NOm53gfpV7OeLCPQfxq
iPwAJKKNaq081Ov/D2XXuL14D1xvex56pM1Ue6EcflCF5/bmkC3UfKhMfrJ7FFQF
6yk0KVs+kAYvVty/NmV8Ng==
=4g4U
-----END PGP MESSAGE-----
```

A partir del cifrado ejecutado, el contenido del archivo original es representado mediante una secuencia de caracteres alfanuméricos

Una vez que los archivos han sido correctamente firmados y cifrados, se pueden eliminar las versiones originales y, posteriormente, recuperar la información a partir de los archivos protegidos.

Para eliminar los archivos originales, se debe ejecutar el siguiente comando:

```
1. rm *.txt *.py
```

Se debe ejecutar nuevamente el comando ls -l con el objetivo de verificar que los archivos originales han sido eliminados correctamente del directorio.

```
1. ls -l
```

```
proyecto@deusmarket:~/python-prueba$ ls -l
total 8
drwxrwxr-x 2 proyecto proyecto 4096 Keg   7 17:30 encriptados
drwxrwxr-x 2 proyecto proyecto 4096 Keg   7 17:22 firmas
```

Ante la ausencia de los archivos originales, se procederá a realizar el proceso de descifrado y validación de los archivos cifrados. Para ello, es necesario acceder al directorio desencriptados y obtener un listado de todos los archivos contenidos en dicha carpeta.

```
1. cd desencriptados/ && ls -l
```

```
proyecto@deusmarket:~/python-prueba$ cd encriptados/ && ls -l
total 12
-rw-rw-r-- 1 proyecto proyecto 736 Keg   7 17:30 test1.txt.gpg
-rw-rw-r-- 1 proyecto proyecto 781 Keg   7 17:30 test2.py.gpg
-rw-rw-r-- 1 proyecto proyecto 736 Keg   7 17:30 test3.txt.gpg
```

Para llevar a cabo el proceso de descifrado de los archivos, se debe ejecutar el script desencriptado desde el directorio actual.

```
proyecto@deusmarket:~/python-prueba/encriptados$ desencriptados
El directorio desencriptados fue creado
ok:  True
status:  decryption ok
stderr:  [GNUPG:] ENC_TO 3A9E3B52EE884B5C 1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] DECRYPTION_KEY D9FDCA3B68CD01F93FB4B9FD3A9E3B52EE884B5C CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 u
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
gpg: encrypted with 3072-bit RSA key, ID 3A9E3B52EE884B5C, created 2023-05-07
      "Deusmarket <ventasonline@deusmarket.pe>"
[GNUPG:] BEGIN_DECRYPTION
[GNUPG:] DECRYPTION_COMPLIANCE_MODE 23
[GNUPG:] DECRYPTION_INFO 2 9 0
[GNUPG:] PLAINTEXT 62 1683498651
[GNUPG:] PLAINTEXT_LENGTH 37
[GNUPG:] DECRYPTION_OKAY
[GNUPG:] GOODMDC
[GNUPG:] END_DECRYPTION
```

La ejecución del script retornó el estado “decryption ok” para el archivo, lo que indica que el proceso de descifrado se realizó correctamente.

A continuación, se debe ingresar al nuevo directorio desencriptados/ para verificar los archivos recuperados. Dentro de esta carpeta, se podrá visualizar el contenido del archivo test1.txt utilizando el comando cat.

```
1. cd desencriptados/ && cat test1.txt
```

```
proyecto@deusmarket:~/python-prueba/encriptados$ cd desencriptados/ && cat test1.txt
 Este es el primer archivo de prueba
```

Se ha logrado recuperar el contenido del archivo test1.txt, el cual había sido eliminado previamente, demostrando la efectividad del proceso de descifrado.

Para validar la autenticidad del mensaje original, se empleará el script verificar_archivo, encargado de comprobar su firma digital.

En la firma digital se incluye la identidad del firmante junto con la función hash generada desde la información del archivo. Durante el proceso de revisión, la herramienta GPG utiliza

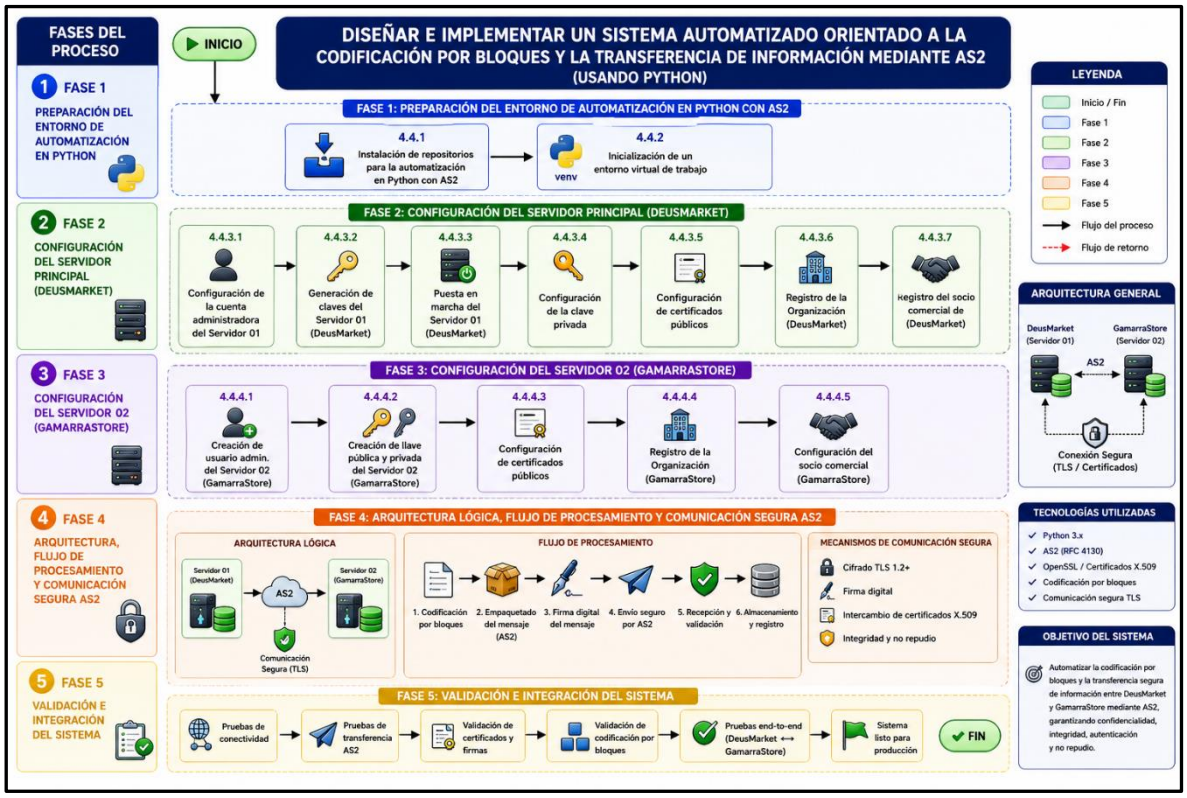
la clave de cifrado pública del emisor en conjunto con un algoritmo de hash para recalcular dicho valor a partir de los datos originales. Para que la validación sea considerada correcta, el hash obtenido debe coincidir exactamente con el valor incluido en la firma.

En caso de que se produzca cualquier alteración en el archivo original, en la firma digital o en la clave pública del remitente, el valor hash resultante será diferente, lo que provocará el fallo del proceso de verificación.

Se debe ejecutar el codigo desde el interior del directorio desencriptados/ para proceder con la verificación correspondiente.

```
proyecto@deusmarket:~/python-prueba/encriptados/desencriptados$ verificar_archivo
test1.txt  signature valid
```

4.4 Diseño e implementación del sistema automatizado



Se procederá a la instalación de Django, el cual permitirá la configuración y gestión de un entorno virtual en Python para el desarrollo del proyecto.

4.4.1 Repositorios para la automatización en Python con AS2

Ejecutaremos el siguiente código para actualizar los paquetes:

```
1. sudo apt update
```

Verificaremos la versión de Python que tenemos instalado:

```
1. python3 -V
```

```
(virtual) proyecto@deusmarket:~/deusmarket$ python -V
Python 3.10.6
```

Realizaremos la instalación de pip desde los repositorios de Ubuntu:

```
1. sudo apt install python3-pip
```

Ahora instalaremos el pip, que nos permite instalar el repositorio venv:

```
1. sudo apt install python3-venv
```

Ahora, crearemos nuestro proyecto en un entorno virtual:

```
1. mkdir ~/deusmarket
2. cd ~/deusmarket
```

4.4.2 Inicialización de un entorno virtual de trabajo

A continuación, se procederá a generar el entorno virtual mediante la ejecución del siguiente comando:

```
1. python3.10 -m venv virtual
```

Con el fin de habilitar el entorno virtual y permitir la instalación de dependencias, es necesario ejecutar la siguiente instrucción:

```
1. source my_env/bin/actívate
```

Para verificar la versión instalada de Django en el sistema, se ejecutará el siguiente comando:

```
(virtual) proyecto@deusmarket:~/deusmarket$ django-admin --version
4.2.1
```

4.4.3 Configuración del Servidor Principal (DeusMarket)

A continuación, se procederá con la creación del proyecto correspondiente.

```
1. (virtual) django-admin startproject deusmarket .
```

```
(virtual) proyecto@deusmarket:~/deusmarket$ ls
db.sqlite3  deusmarket  manage.py  messages  _P1_private.pem  P1_public.pem
```

Se ingresará al directorio DeusMarket con el propósito de realizar las modificaciones necesarias dentro del entorno del proyecto.

```
(virtual) proyecto@deusmarket:~/deusmarket/deusmarket$ ls
asgi.py  __init__.py  __pycache__  settings.py  urls.py  wsgi.py
(virtual) proyecto@deusmarket:~/deusmarket/deusmarket$ nano settings.py
```

En el archivo settings.py, se procederá a añadir la configuración correspondiente del *host*. Asimismo, para habilitar la ejecución del protocolo AS2 dentro del sistema, se incorporará la dependencia "pyas2" en la configuración del proyecto.

```
ALLOWED_HOSTS = ['127.0.0.1']


 # Application definition

 INSTALLED_APPS = [
     'django.contrib.admin',
     'django.contrib.auth',
     'django.contrib.contenttypes',
     'django.contrib.sessions',
     'django.contrib.messages',
     'django.contrib.staticfiles',
    'pyas2',
]
```

Posteriormente, se accederá al archivo urls.py con la finalidad de incorporar las configuraciones necesarias, agregando las rutas correspondientes para el funcionamiento del sistema.

```
from django.contrib import admin
from django.urls import include
from django.urls import path

urlpatterns = [
    path('admin/', admin.site.urls),
    path('pyas2/', include('pyas2.urls')),
```

A continuación, se verificará que no existan restricciones en el firewall que impidan el correcto funcionamiento del sistema. Para ello, se habilitará el puerto 8000, el cual será utilizado para el acceso a la interfaz web de la aplicación.

```
1. sudo ufw allow 8000
```

```
(virtual) proyecto@deusmarket:~/deusmarket/deusmarket$ sudo ufw allow 8000
[sudo] password for proyecto:
Skipping adding existing rule
Skipping adding existing rule (v6)
(virtual) proyecto@deusmarket:~/deusmarket/deusmarket$
```

4.4.3.1 Configuración de la cuenta administradora del Servidor 01

Se procederá a la creación del usuario administrador, el cual permitirá el acceso y gestión del servidor, mediante la ejecución del comando correspondiente.

```
1. python manage.py createsuperuser
```

```
(virtual) proyecto@deusmarket:~/deusmarket$ python manage.py createsuperuser
Username (leave blank to use 'proyecto'): deusmarket
Email address: ventasonline@deusmarket.pe
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
(virtual) proyecto@deusmarket:~/deusmarket$
```

4.4.3.2 Generación de claves del Servidor 01 (DeusMarket)

A continuación, se procederá a crear el par de claves criptográficas pública y privada correspondiente.

```
 1. $ openssl req -x509 -newkey rsa:2048 -sha256 -keyout
P1_private.pem -out P1_public.pem -days 365
 2. Generating a 2048 bit RSA private key
 3. .....+++
 4.
.....
.....+++
 5. writing new private key to 'P1_private.pem'
 6. Enter PEM pass phrase:
 7. Verifying - Enter PEM pass phrase:
 8. -----
 9. You are about to be asked to enter information that will be
incorporated
10. into your certificate request.
11. What you are about to enter is what is called a Distinguished
Name or a DN.
12. There are quite a few fields but you can leave some blank
13. For some fields there will be a default value,
14. If you enter '.', the field will be left blank.
15. -----
16. Country Name (2 letter code) [AU]:PE
17. State or Province Name (full name) [Some-State]:Lima
18. Locality Name (eg, city) []:Lima
19. Organization Name (eg, company) [Internet Widgits Pty
Ltd]:Deusmarket
20. Organizational Unit Name (eg, section) []:Deusmarket
21. Common Name (e.g. server FQDN or YOUR name) []:Deusmarket
22. Email Address []: ventasonline@deusmarket.pe
23. $ cat P1_public.pem >> P1_private.pem
```

4.4.3.3 Puesta en marcha del Servidor 01 (DeusMarket)

A continuación, se procederá a la ejecución del Servidor 01 correspondiente a la plataforma DeusMarket, mediante el comando establecido para su inicialización.

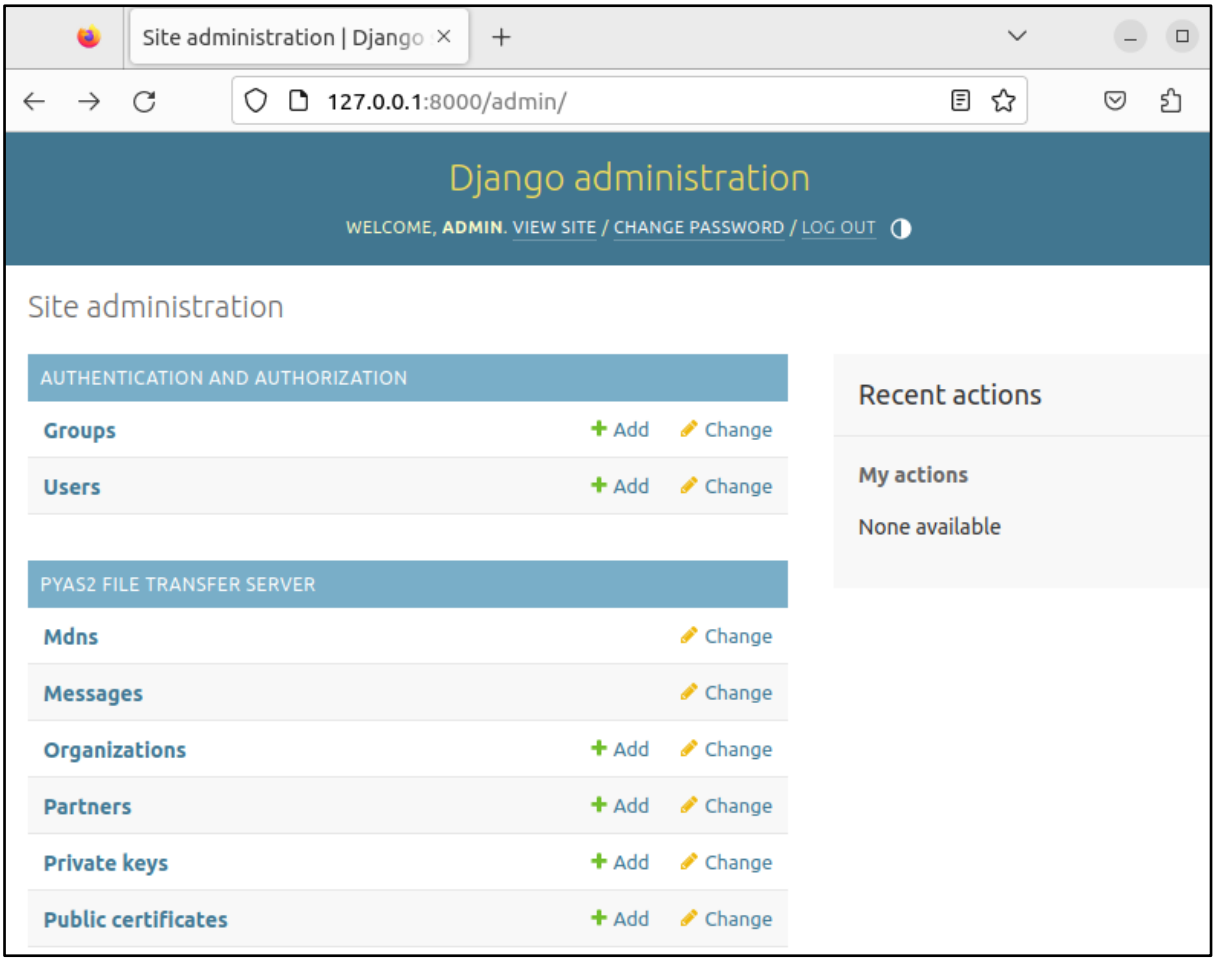
```
1. python manage.py runserver
```

```
(virtual) proyecto@deusmarket:~/deusmarket$ python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
May 13, 2023 - 22:35:43
Django version 4.2.1, using settings 'deusmarket.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

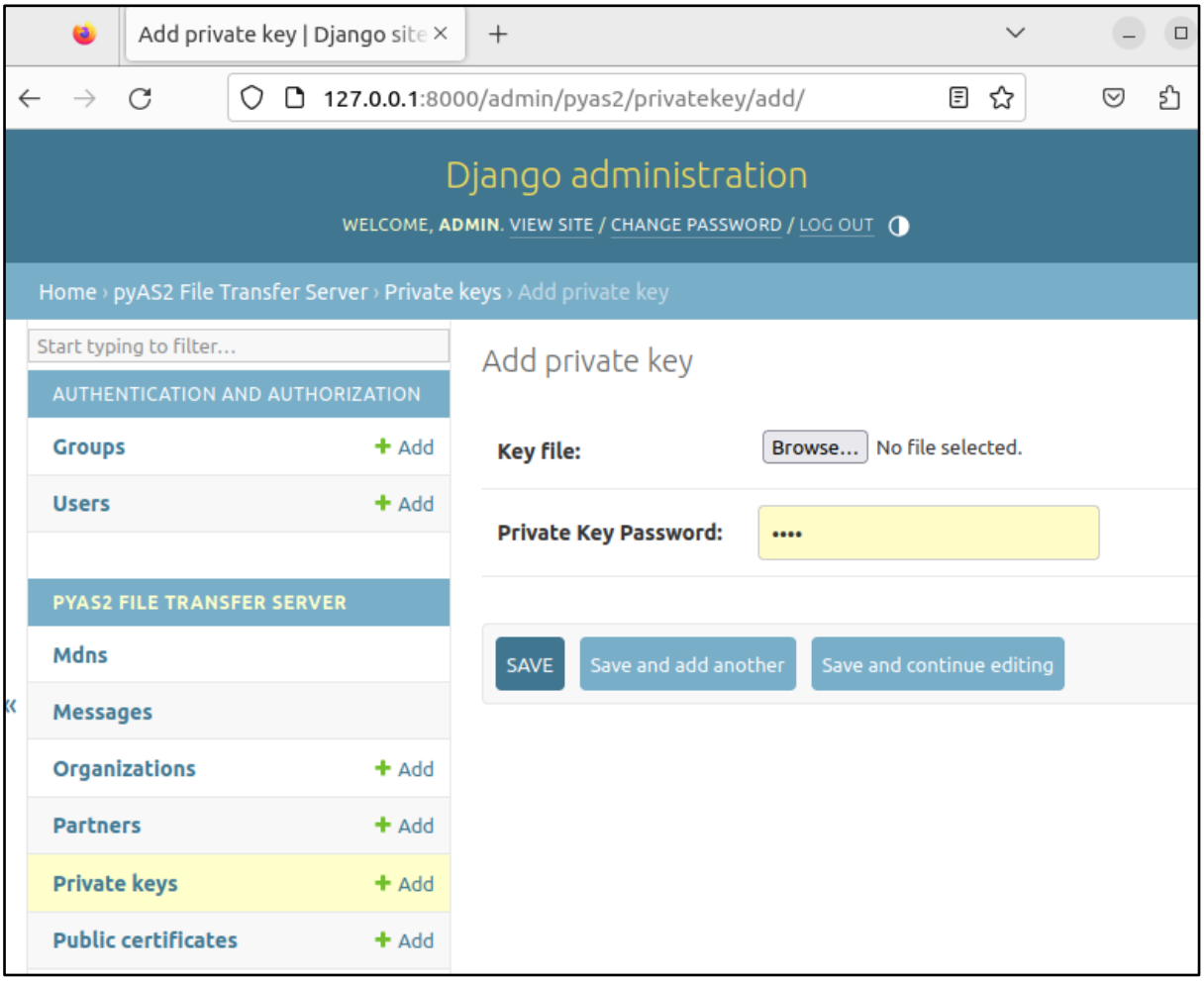
Se accederá al servicio de envío de mensajes mediante AS2 a través de la dirección 127.0.0.1:8000/admin/pyas2/, donde se gestionará la interfaz administrativa del sistema.

Figura 37
Evidencia del servicio de mensajería AS2 en funcionamiento.



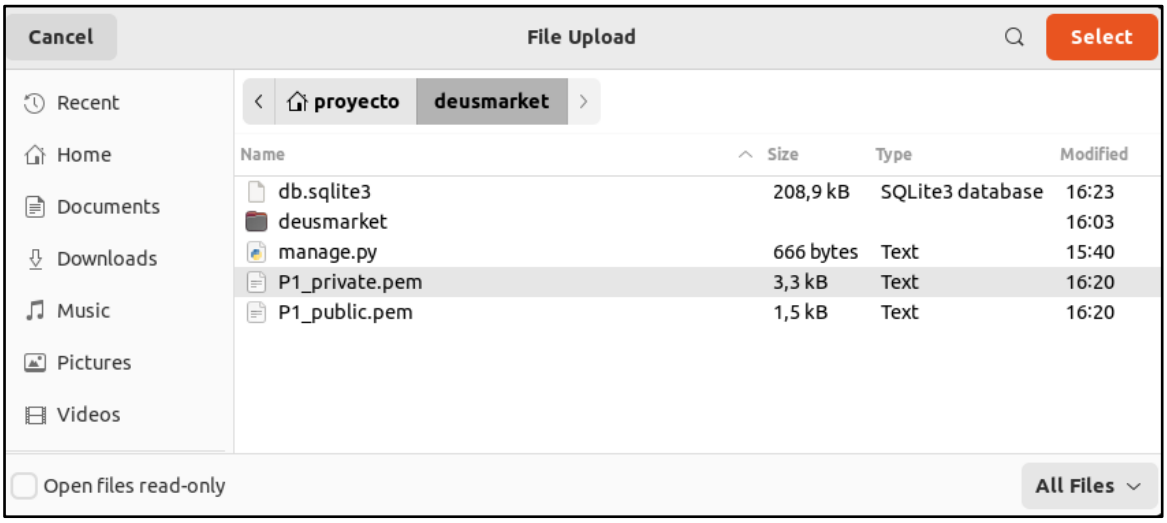
4.4.3.4 Configuración de la clave privada

Figura 38
Evidencia de la asignación de la clave privada en el sistema.



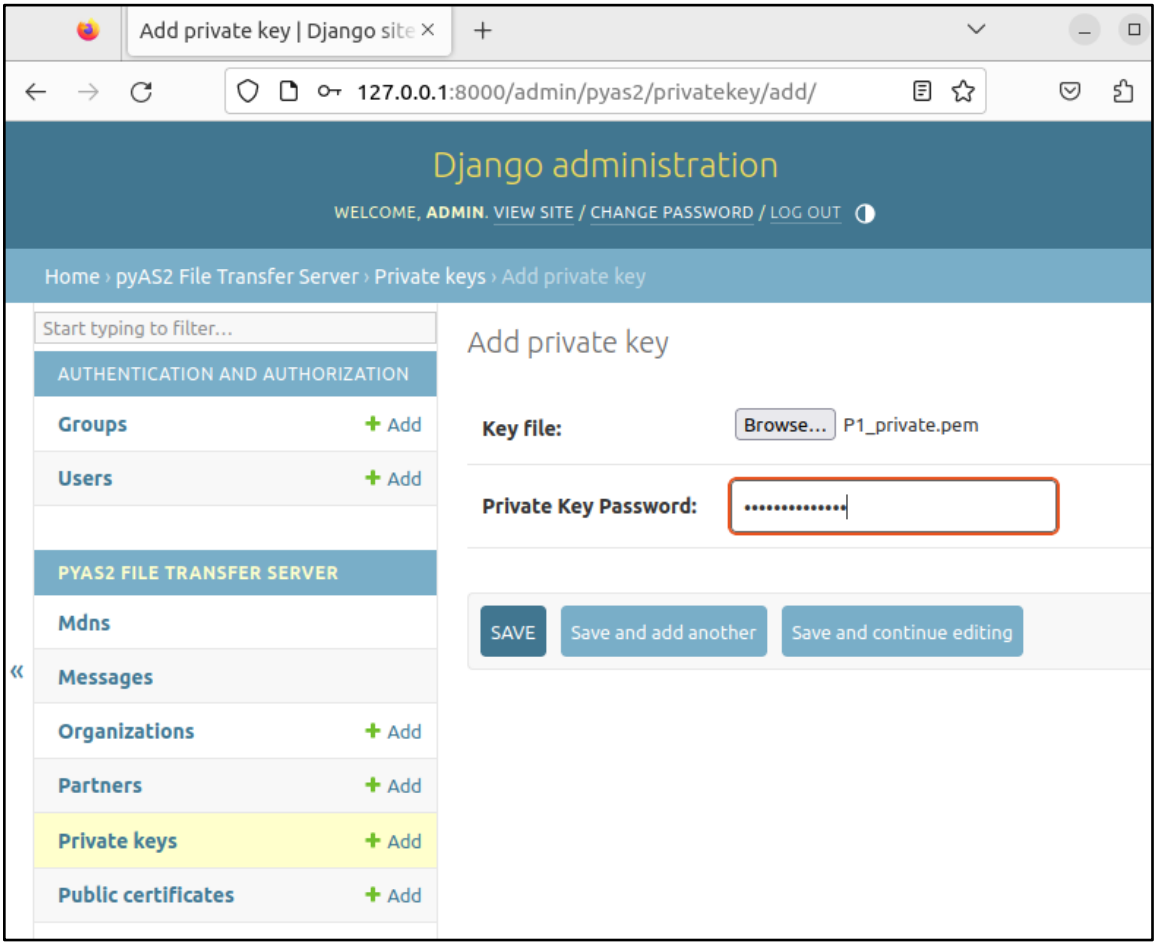
Se procederá a seleccionar la clave privada P1_private.pem, la cual fue generada previamente en el sistema.

Figura 39
Selección del archivo PEM previamente generado.



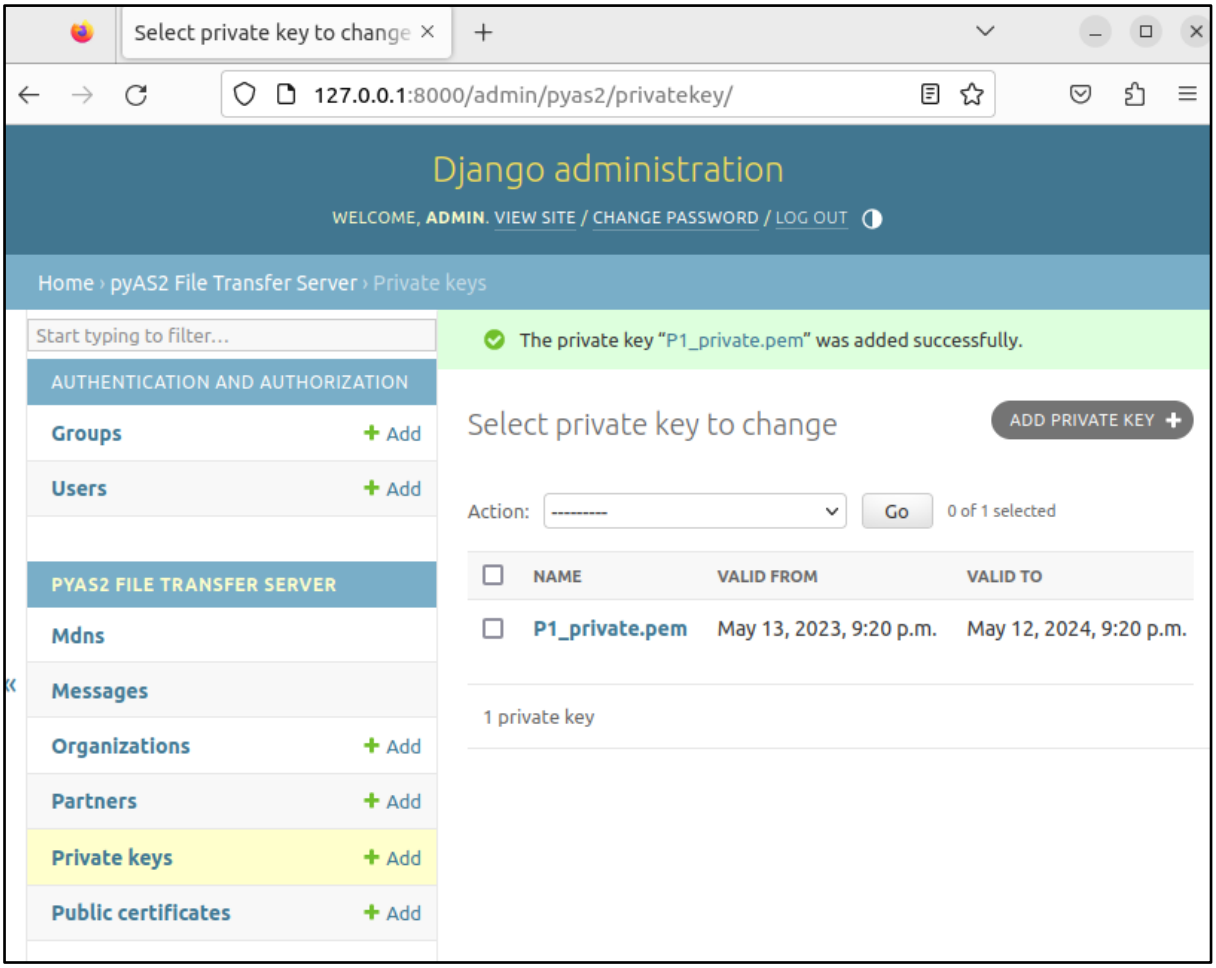
Se ingresará la contraseña establecida para este caso, correspondiente a “deusmarket2023”, con el fin de autenticar el acceso a la clave configurada.

Figura 40
Evidencia de la selección de la clave configurada en el sistema.



Se observa que la clave privada ha sido generada y configurada de manera exitosa.

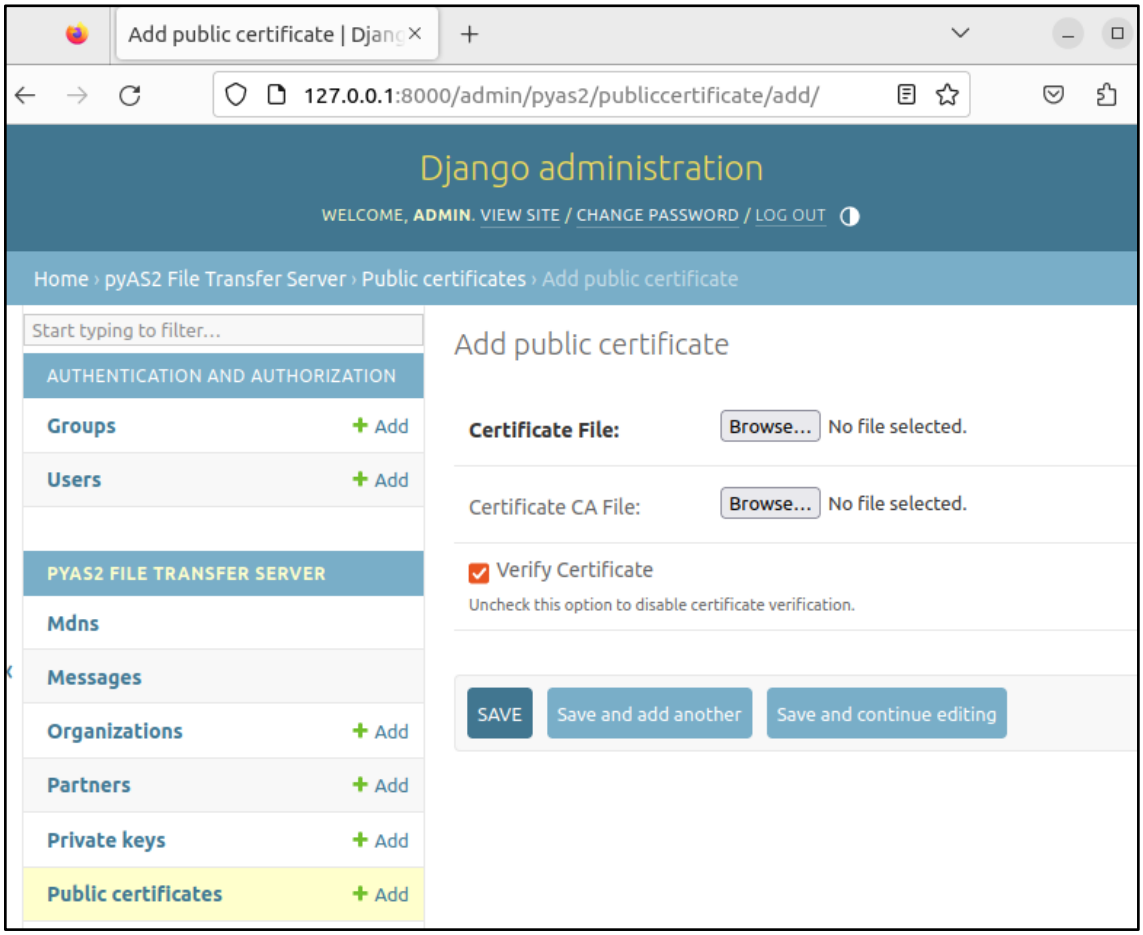
Figura 41
Evidencia de la generación exitosa de la clave privada.



4.4.3.5 Configuración de certificados públicos

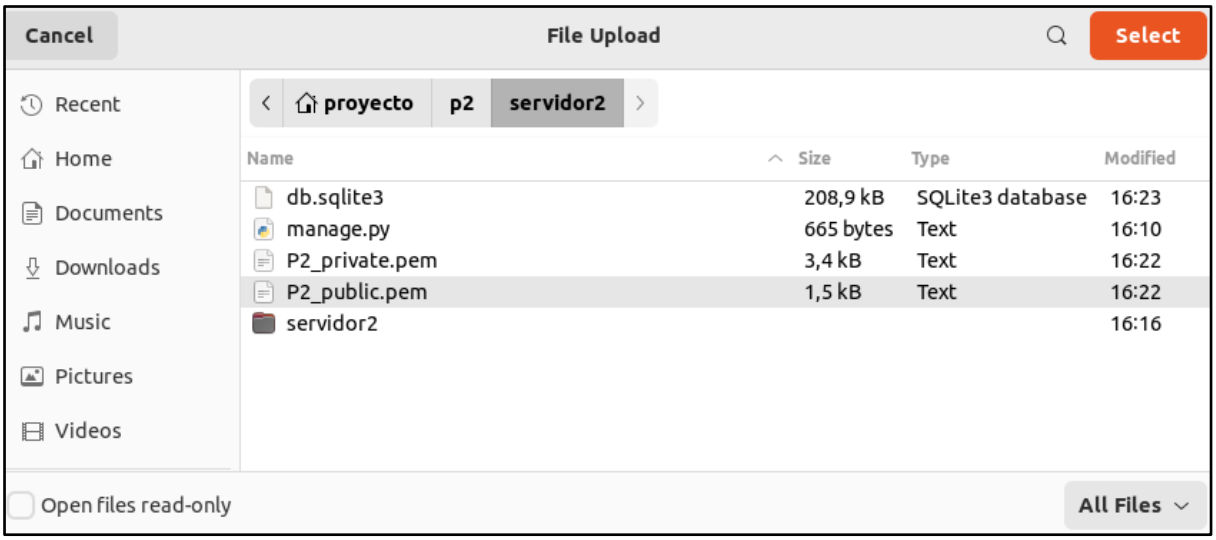
A continuación, se procederá al acceso a la sección de certificados públicos del sistema.

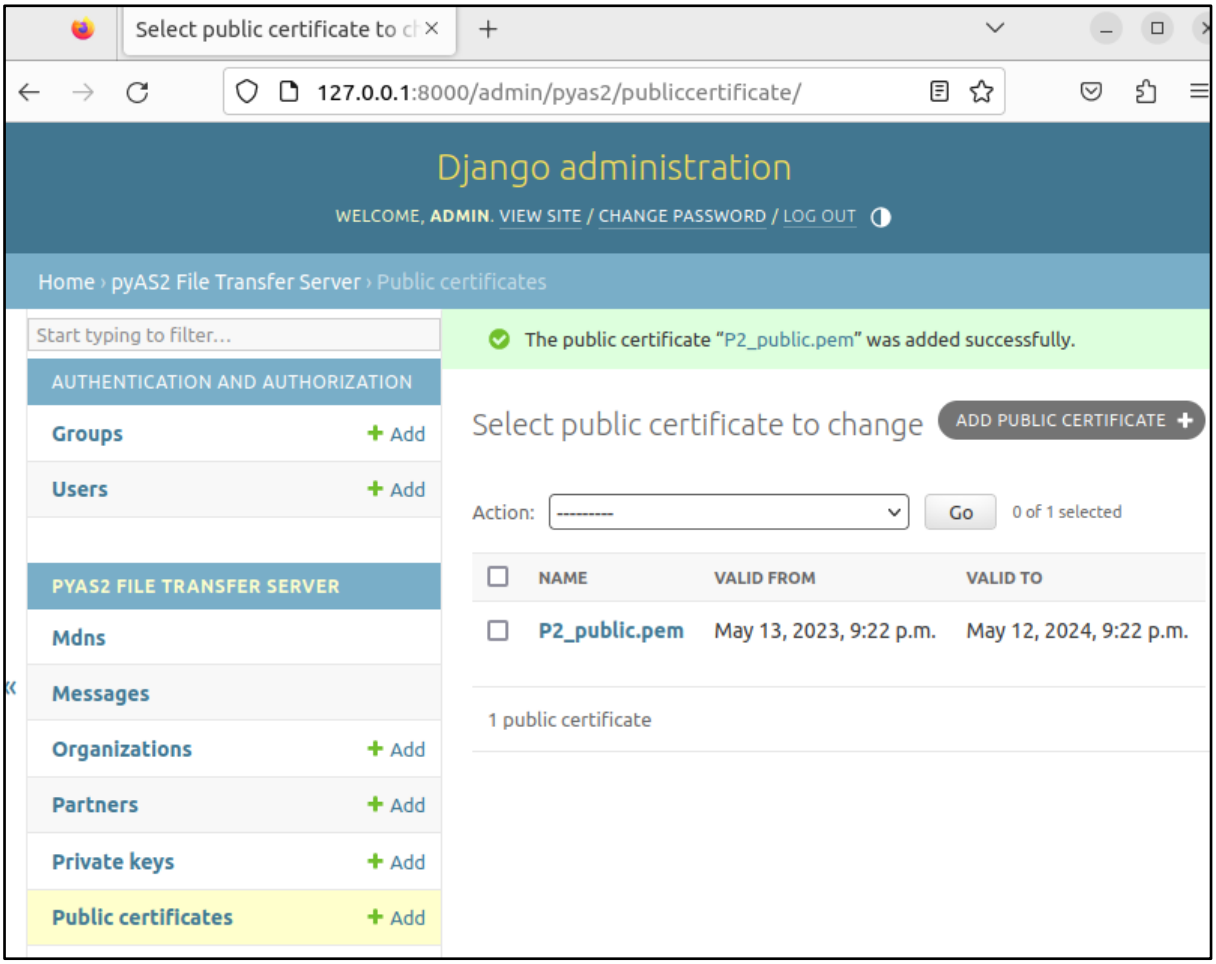
Figura 42
Evidencia de la configuración de los certificados públicos en el sistema.



Se procederá a seleccionar la clave pública P2_public.pem, previamente generada, para su correspondiente configuración en el sistema.

Figura 43
Selección de la clave pública P2.



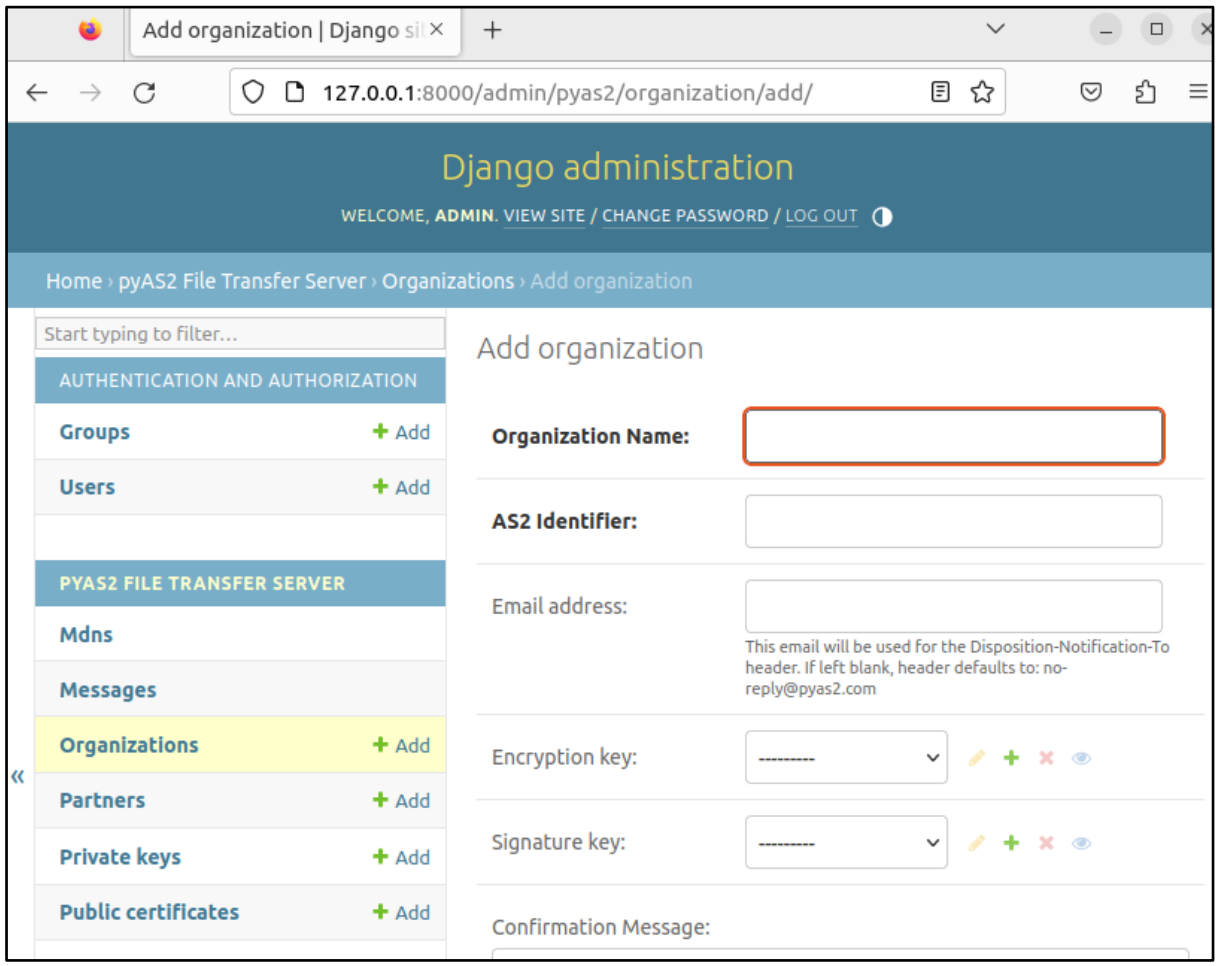


4.4.3.6 Registro de la Organización (DeusMarket)

Se procederá a la creación del registro de la organización correspondiente a DeusMarket, ingresando los datos requeridos en la configuración del sistema:

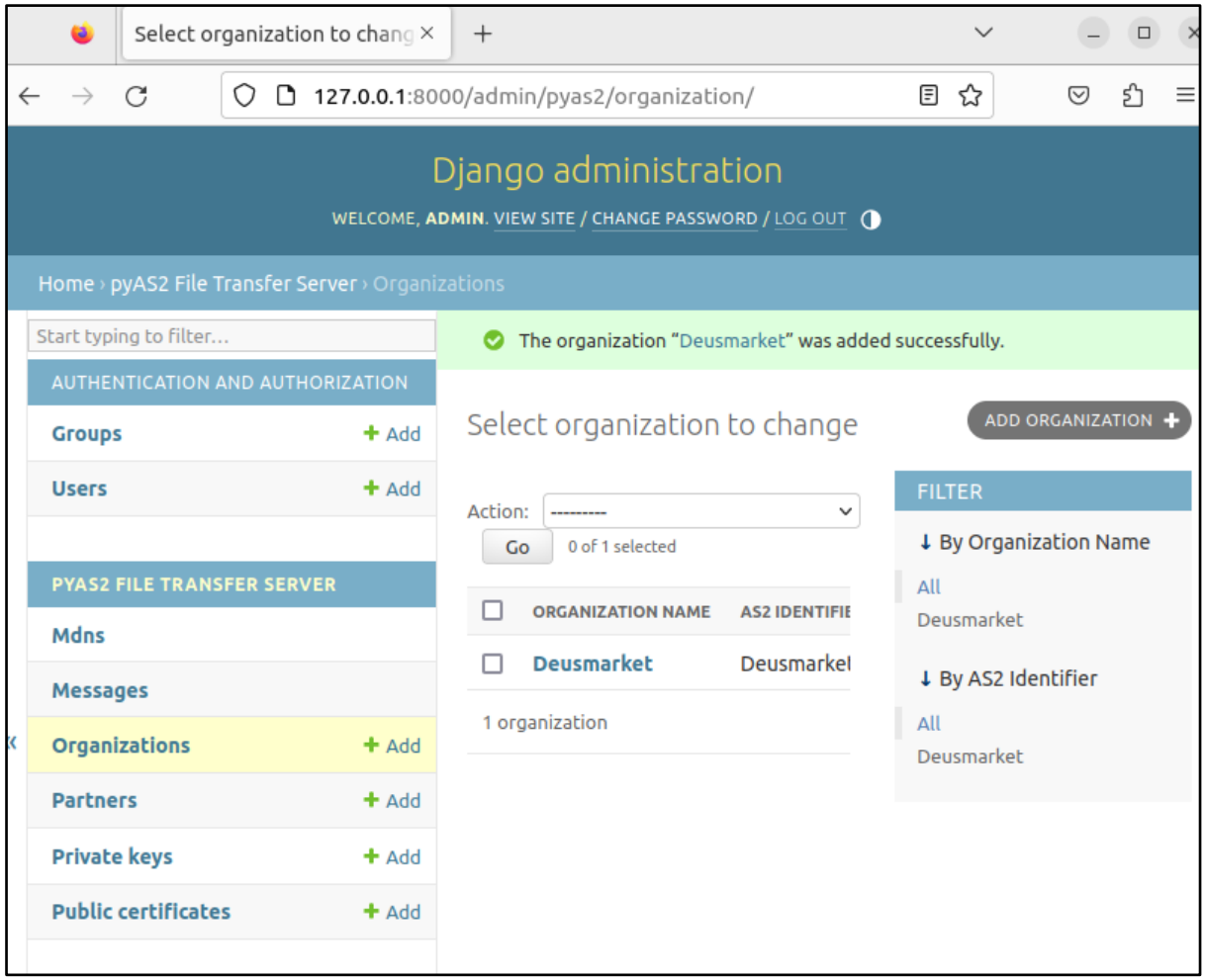
- **Nombre de la organización:** DeusMarket
- **Identificador AS2:** DeusMarket
- **Clave de cifrado:** P1_private.pem
- **Clave de firma:** P1_private.pem

Figura 44
Creación de Organización DEUSMARKET



Se observa que la organización ha sido creada de manera satisfactoria dentro del sistema.

Figura 45
Evidencia del registro exitoso de la organización en el sistema.



4.4.3.7 Registro del socio comercial de (DeusMarket)

A continuación, se procederá a la creación del socio comercial dentro del sistema DeusMarket, ingresando la información correspondiente:

Nombre del partner: GamarraStore

Identificador AS2: GamarraStore

URL de destino: http://127.0.0.1:8001/pyas2/as2receive

Cifrado del mensaje: 3DES

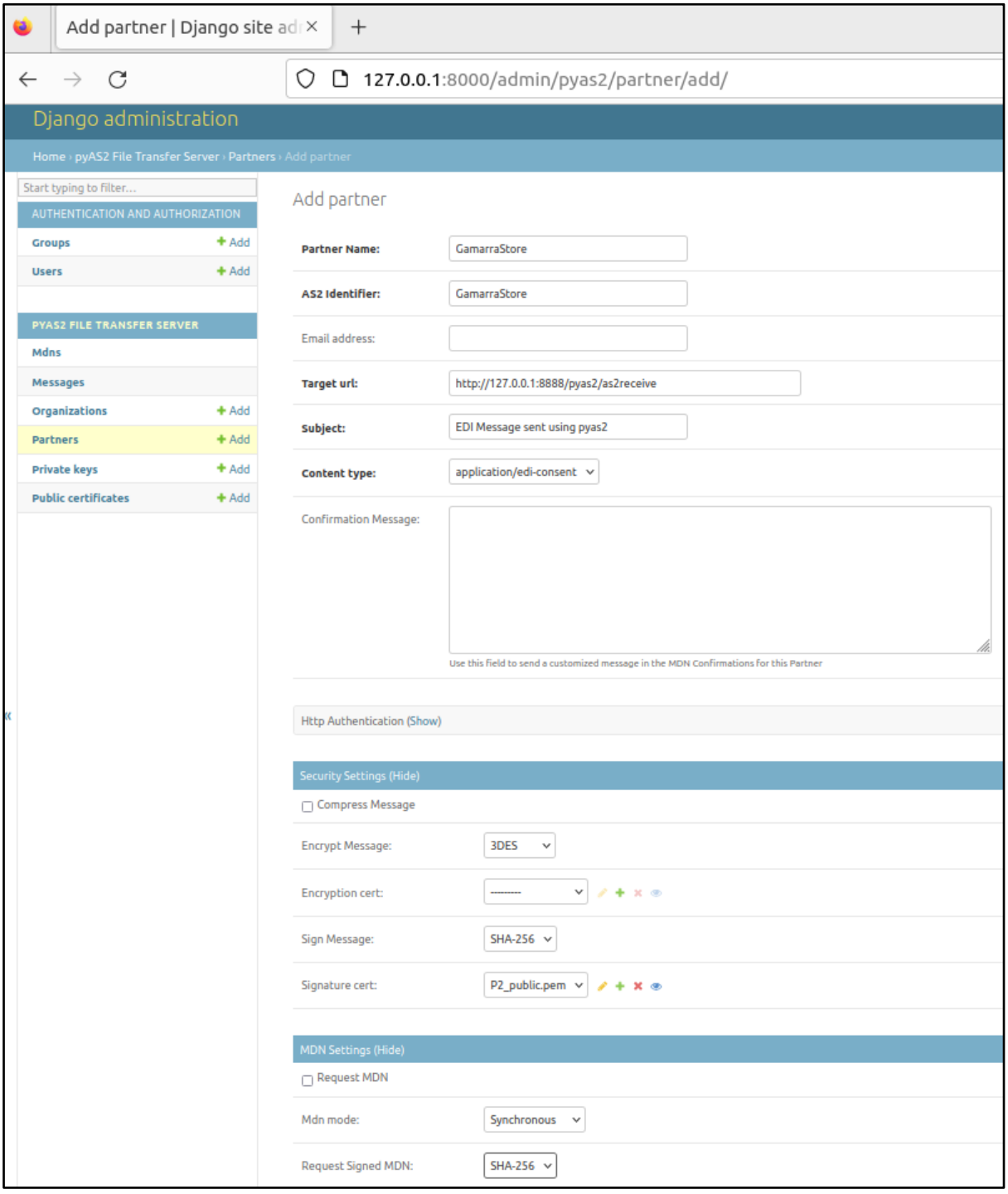
Firma del mensaje: SHA-256

Certificado de firma: P2_public.pem

Modo MDN: Síncrono

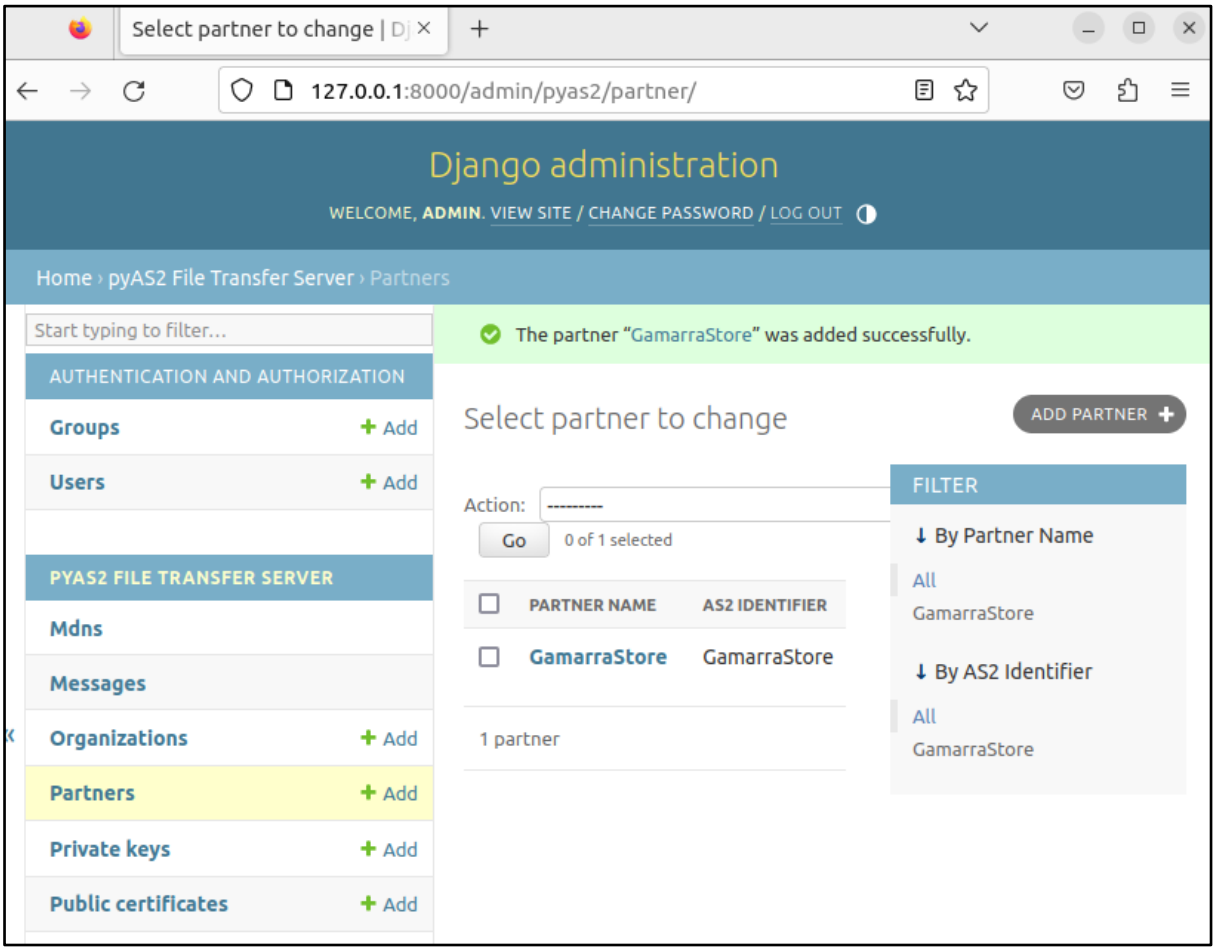
Solicitud de MDN firmado: SHA-256

Figura 46
Creación del socio comercial



De igual forma, se procederá a validar que el partner fue generado de manera exitosa.

Figura 47
Evidencia del registro del socio comercial (Partner) en el sistema.



4.4.4 Configuración del Servidor 02 (GamarraStore)

4.4.4.1 Creación de usuario administrador del Servidor 02 (GamarraStore)

Realizaremos la misma configuración con respecto al Servidor 01 hasta la configuración de la creación del usuario administrador

```
1. proyecto@deusmarket:~/p2/servidor2$ python manage.py
createsuperuser
2. Username (leave blank to use 'proyecto'): admin
3. Email address: ventasgamarra2023@gmail.com
4. Password:
5. Password (again):
6. This password is too short. It must contain at least 8
characters.
7. Bypass password validation and create user anyway? [y/N]: y
8. Superuser created successfully.
```

4.4.4.2 Creación de llave pública y privada del Servidor 02 (GamarraStore)

Configuraremos la llave privada con el siguiente comando:

```
 1.  proyecto@deusmarket:~/p2/servidor2$ openssl req -x509 -newkey
rsa:2048 -sha256 -keyout P2_private.pem -out P2_public.pem -days
365
 2.
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*
.+......+......*.++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++
 3.
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*
..+.....+....+.............+........+........+...+..........+
...++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++
 4.  Enter PEM pass phrase:
 5.  Verifying - Enter PEM pass phrase:
 6.  -----
 7.  You are about to be asked to enter information that will be
incorporated
 8.  into your certificate request.
 9.  What you are about to enter is what is called a Distinguished
Name or a DN.
10.  There are quite a few fields but you can leave some blank
11.  For some fields there will be a default value,
12.  If you enter '.', the field will be left blank.
13.  -----
14.  Country Name (2 letter code) [AU]:IN
15.  State or Province Name (full name) [Some-State]:Arequipa
16.  Locality Name (eg, city) []:Arequipa
17.  Organization Name (eg, company) [Internet Widgits Pty
Ltd]:GamarraStore
18.  Organizational Unit Name (eg, section) []:GamarraStore
19.  Common Name (e.g. server FQDN or YOUR name) []:GamarraStore
20.  Email Address []:ventasgamarra2023@gmail.com
```

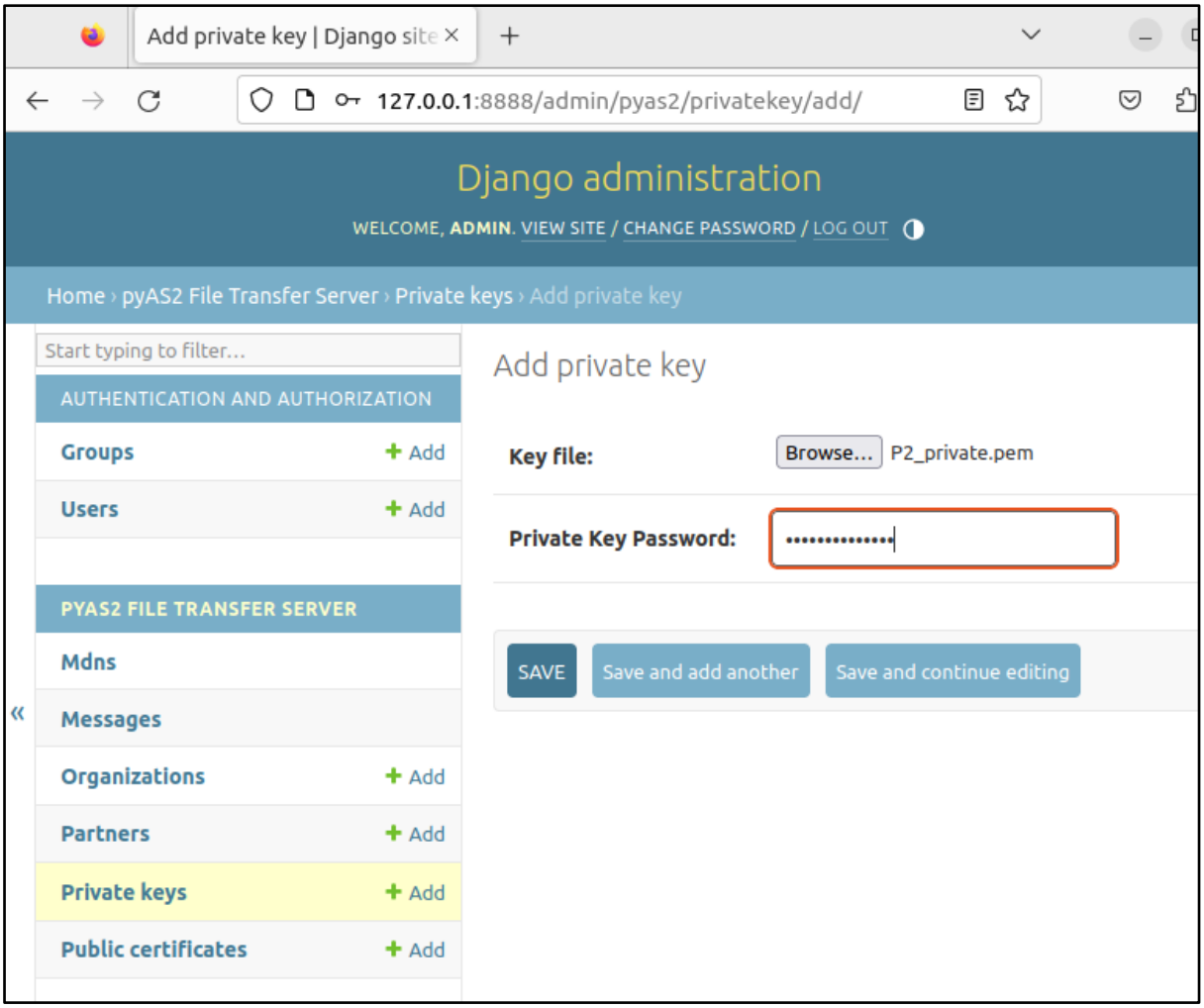
Ejecución del Servidor 02 (GamarraStore)

```
1. (p2) proyecto@deusmarket:~/p2/servidor2$ python manage.py
runserver 8888
2. Watching for file changes with StatReloader
3. Performing system checks...
4.
5. System check identified no issues (0 silenced).
6. May 13, 2023 - 21:17:35
7. Django version 4.2.1, using settings 'servidor2.settings'
8. Starting development server at http://127.0.0.1:8888/
```

Asignación de llave privada

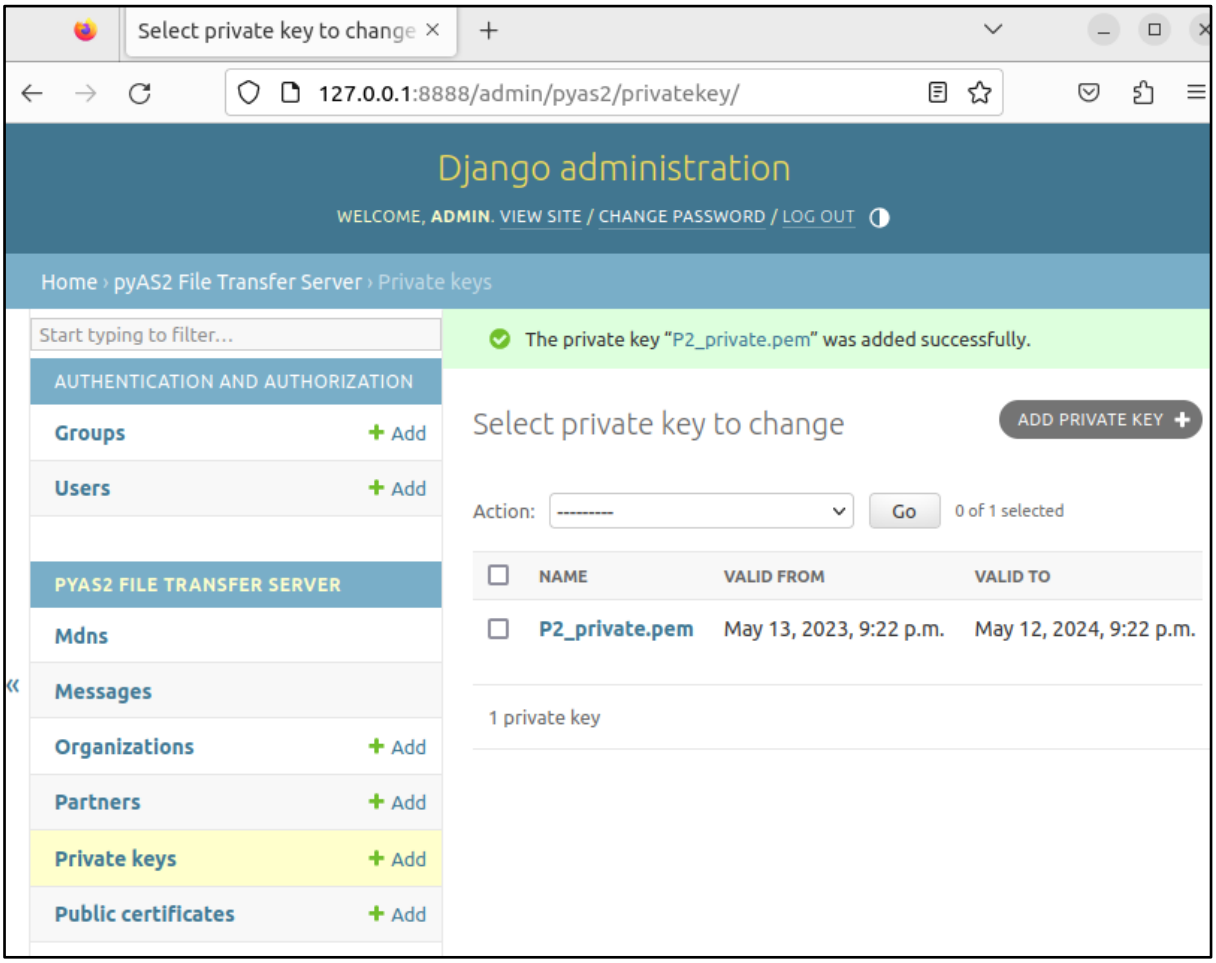
Crearemos nuestra llave privada, seleccionaremos el archivo P2_private.pem

Figura 48
Evidencia de creación de llave privada



Se observa que la clave privada ha sido generada y configurada de manera satisfactoria dentro del sistema.

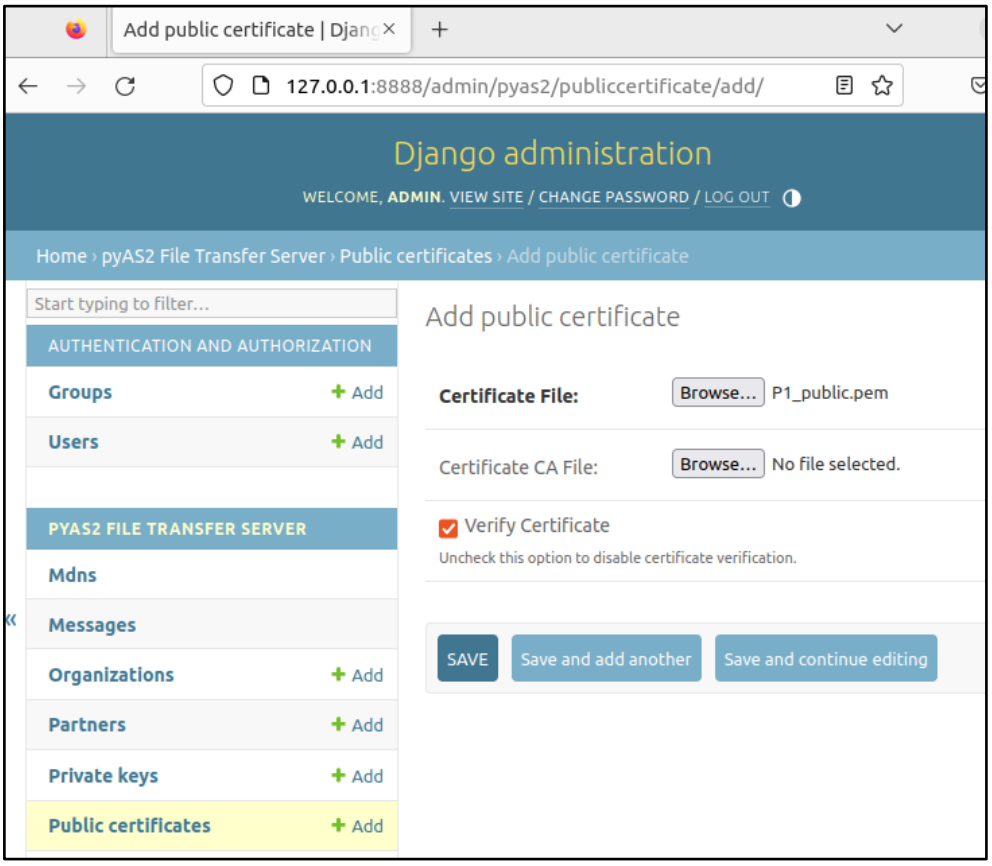
Figura 49
Evidencia de la generación exitosa de la clave privada en el sistema.



4.4.4.3 Configuración de certificados públicos

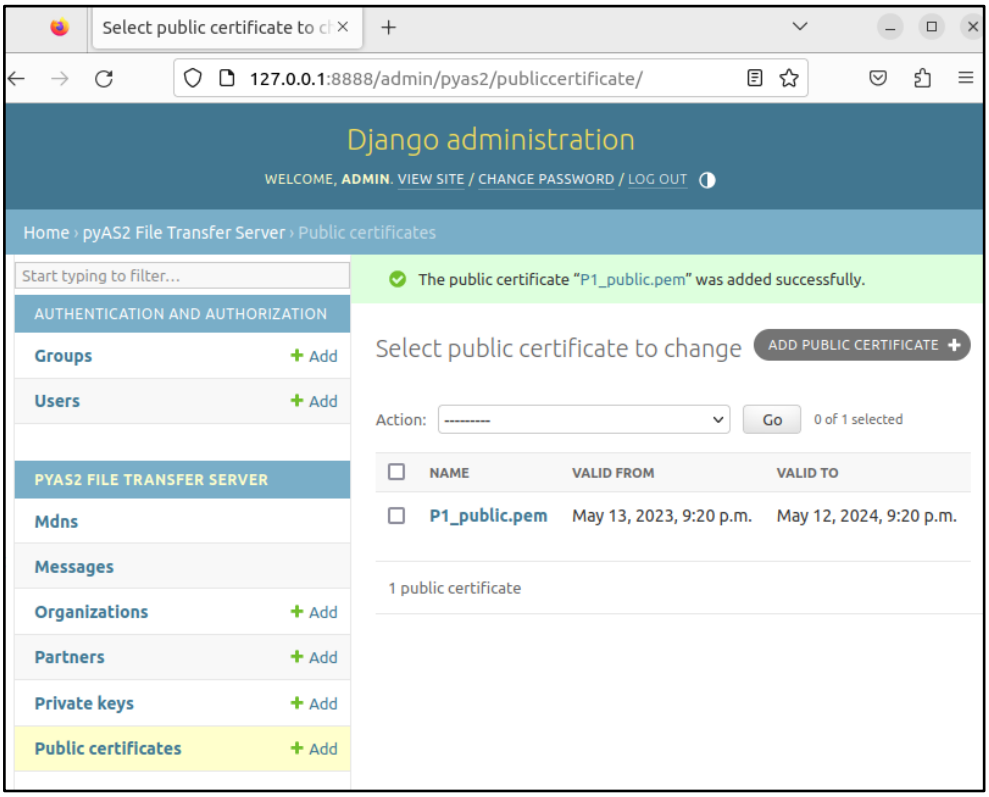
A continuación, se procederá a la incorporación del certificado público en el sistema, utilizando el archivo P1_public.pem para su correcta configuración.

Figura 50
Evidencia de asignación de certificados públicos



Se observa que el certificado público ha sido creado de manera satisfactoria dentro del sistema.

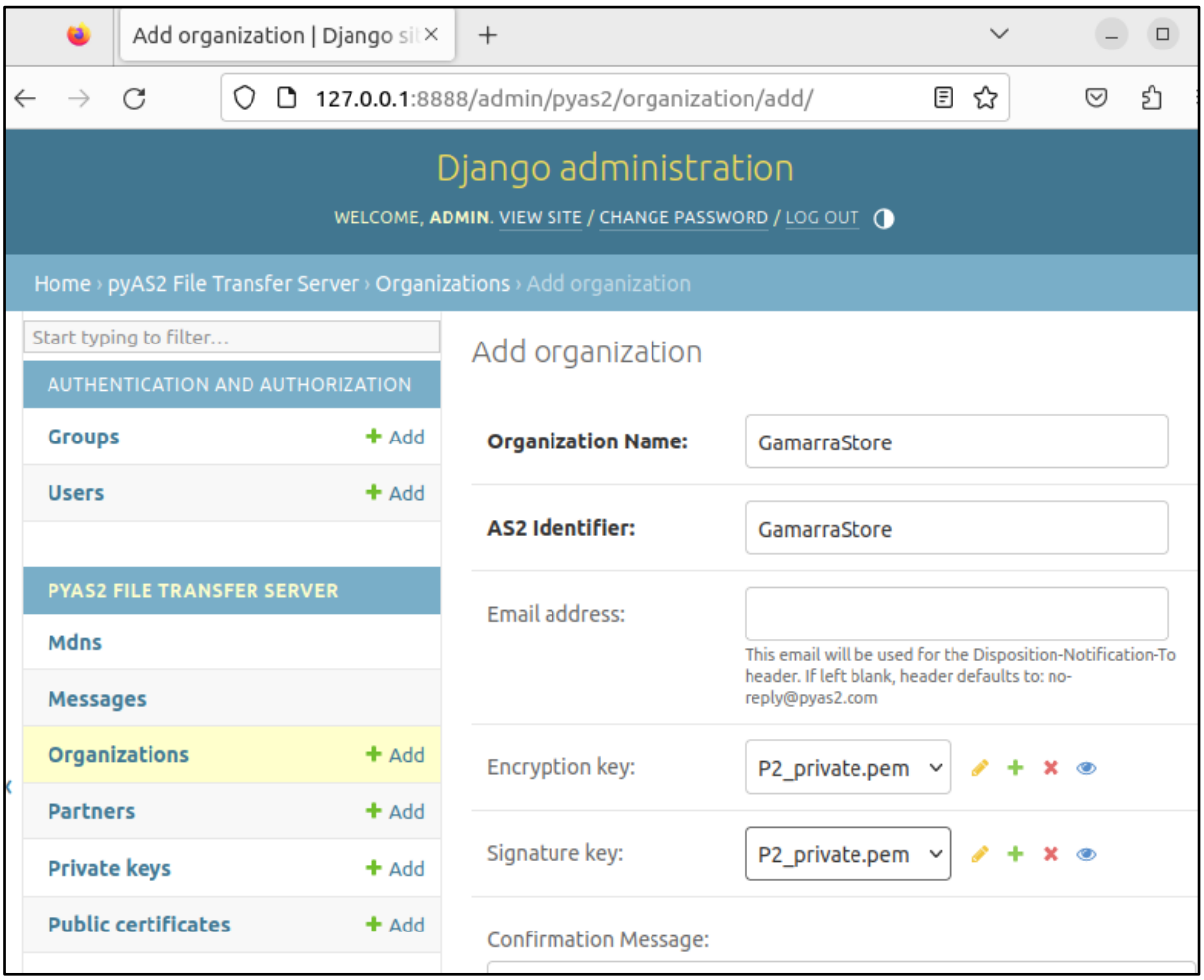
Figura 51
Evidencia de la configuración del certificado público.



4.4.4.4 Registro de la Organización (GamarraStore)

Se procederá a la creación del registro de la organización correspondiente a GamarraStore dentro del sistema, con la finalidad de configurar sus parámetros de identificación y comunicación en el entorno AS2.

Figura 52
Registro de la organización GamarraStore



Nombre de la organización: GamarraStore

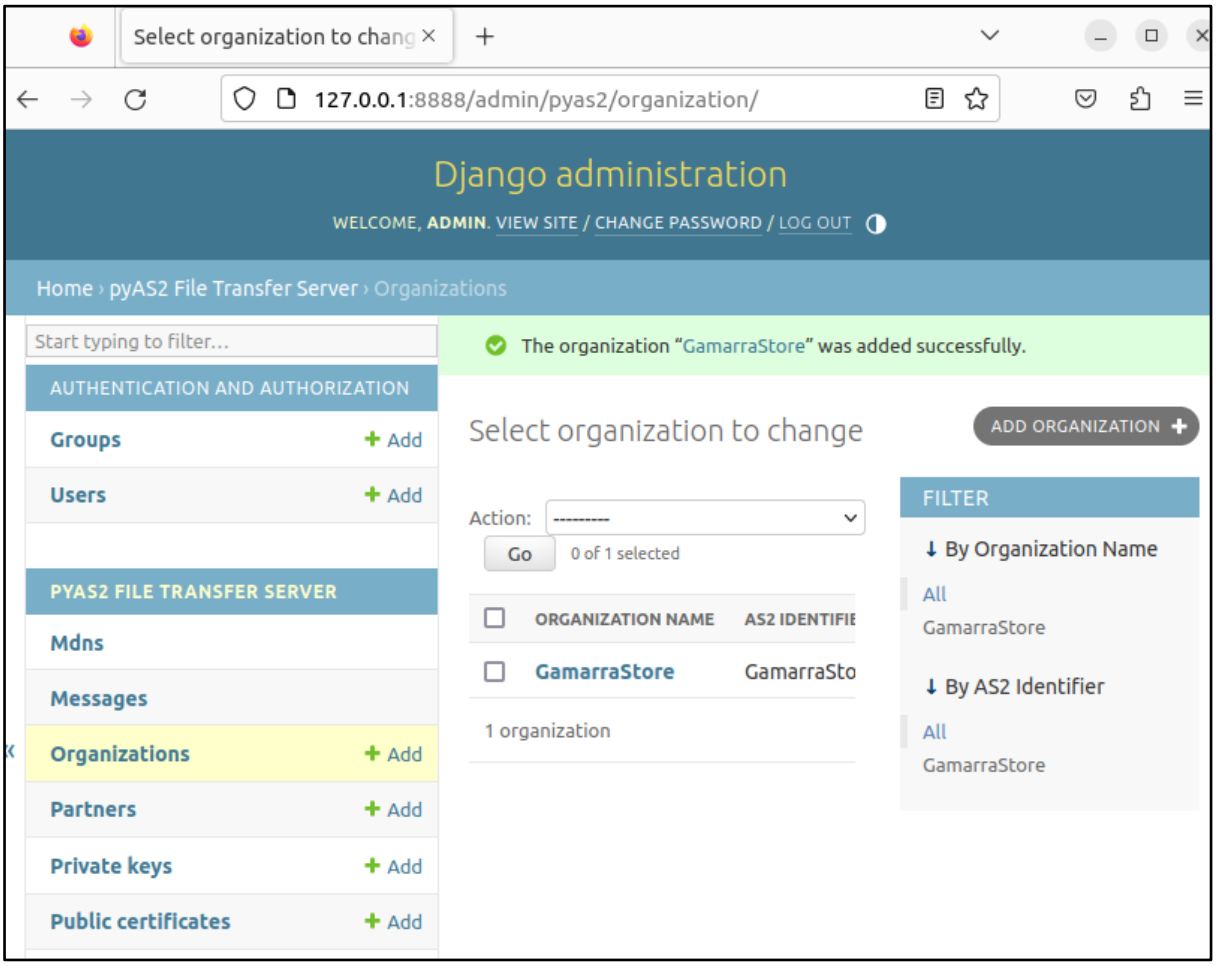
Identificador AS2: GamarraStore

Clave de cifrado: P2_private.pem

Clave de firma: P2_private.pem

De esta forma, la organización ha sido registrada exitosamente dentro del sistema.

Figura 53
Evidencia del registro exitoso de la organización dentro del sistema.

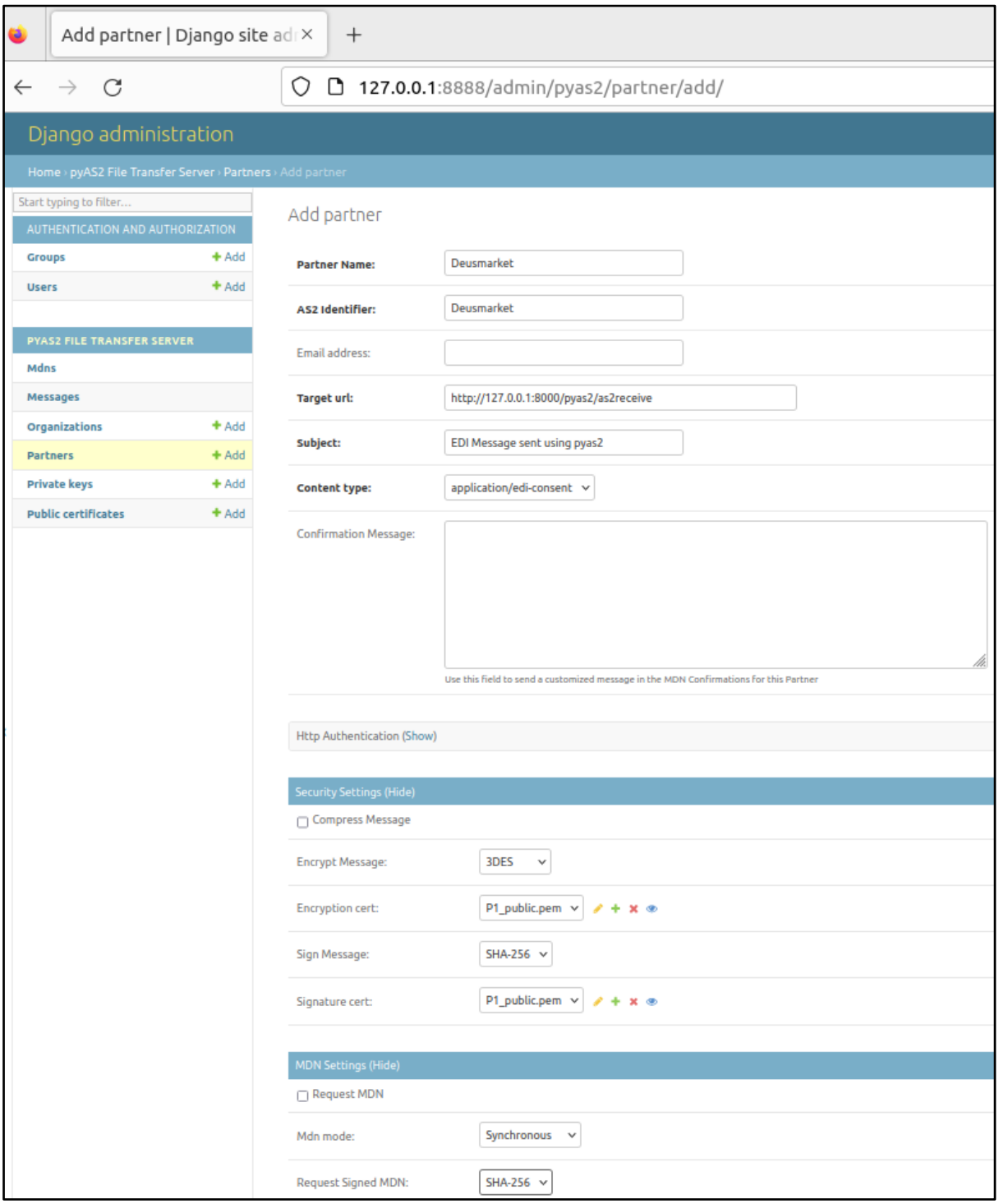


4.4.4.5 Configuración del socio comercial (GamarraStore)

Se procederá al registro del socio comercial GamarraStore dentro del sistema, con el fin de establecer los parámetros necesarios para habilitar la comunicación mediante AS2 en el entorno configurado.

Se establece el cifrado del mensaje mediante 3DES, utilizando SHA-256 para la firma. El certificado de firma corresponde a *P1_public.pem*, y se solicita un MDN firmado con SHA-256.

Figura 54
Evidencia del registro del socio comercial (Partner) en el sistema.



5. CAPÍTULO 5: VALIDACION DE LA SOLUCION

Elaboráremos un plan de pruebas para cada objetivo específico establecido, donde se definirán los tipos de pruebas a llevar a cabo y se establecerá la validación de cumplimiento de las principales funcionalidades. Posteriormente, se realizará un análisis de los resultados obtenidos para verificar si se ha logrado lo propuesto en el proyecto, utilizando los indicadores definidos.

Para detectar fallos potenciales en la plataforma en desarrollo, se llevó a cabo el desarrollo de las pruebas con requisitos ínfimos para su funcionamiento. Esto permitió corregir los posibles errores antes de la construcción final. Además, se utilizará la información obtenida de las pruebas efectuadas para determinar si estas cumplen con los requerimientos establecidos por los stakeholders y si se obtienen el resultado esperado.

Cada ejecución de pruebas incluirá un total de cuatro (4) pruebas, las cuales se llevarán a cabo en un entorno con los medios necesario, luego podremos evaluar los resultados de la evaluación, nos guiaremos de la matriz siguiente.

5.1 Plan de Pruebas

La finalidad del plan de prueba es demostrar cuantitativa y cualitativamente el cumplimiento de los objetivos específicos y se ha logrado las métricas esperaras en el Capítulo 1.

Tabla 12
Tabla de Objetivos Estratégicos

Objetivo Específico	Indicador de Logro	Métrica
Analizar los riesgos asociados al proceso de envío y almacenamiento de mensajes SMTP en una empresa de e-commerce del sector textil, identificando el 100% de los activos, amenazas y vulnerabilidades relevantes, en un plazo de 3 semanas.	- Inventario de activos. - Matriz de Riesgos.	- Porcentaje de número de activos en riesgo. - Porcentaje de riesgos críticos.

Diseñar el sistema de criptografía asimétrica para el envío seguro de mensajes SMTP, especificando los algoritmos, la gestión de claves y los mecanismos de cifrado y firma digital, en un plazo de 4 semanas.	- Arquitectura de proceso de operación de envió y almacenamiento de datos.	Disminución en porcentaje de horas de trabajo manual.
Diseñar el modelo funcional del sistema, incluyendo los procesos de codificación, envío y almacenamiento de mensajes, mediante diagramas y especificaciones técnicas, en un plazo de 4 semanas.	- Programación de script en Python.	Eficiencia del script en Python, porcentaje de aumento de archivos cifrados.
Diseñar e implementar una solución automatizada para procesamiento por bloques y la transferencia de datos vía AS2 con implementación en Python, definiendo su arquitectura lógica, flujo de procesamiento y mecanismos de comunicación segura, en un plazo de 4 semanas.	- Sistema de automatización en AS2. - Resultado de fases luego de la creación de mecanismos de encriptación de datos.	Disminución en porcentaje de horas de trabajo manual.

5.2 Propósito de la Evaluación

El objetivo de las pruebas es definir que tantos riesgos claves afectan la transmisión y el almacenamiento de datos críticos para el negocio creados por herramientas de automatización de oficinas que son accesibles para todos los usuarios y se entregan a los usuarios finales con fácil acceso y centralización.

5.3 Proceso de preparación de validación

El encargado de Sistemas liderará la etapa inicial de las pruebas, donde su tarea consistirá en descubrir los datos de mayor importancia y evaluar los riesgos más significativos asociados a su transmisión y almacenamiento.

Estos riesgos se clasificarán y ordenarán según el enfoque establecido, y se llevará a cabo una encuesta dirigida a los usuarios involucrados, quienes tienen conocimiento.

5.3.1 Elaboración de prueba Objetivo E1

Objetivo 1: Analizar los riesgos asociados al proceso de envío y almacenamiento de mensajes SMTP en una empresa de e-commerce del sector textil, identificando el 100% de los activos, amenazas y vulnerabilidades relevantes, en un plazo de 3 semanas.

Tabla 13
Tabla de preparación de Objetivo Estratégico 1

<table>
<tr><td colspan="3">Plan de Prueba</td></tr>
<tr><td colspan="3">Objetivo: Validar la implementación de controles de autenticación en el servicio de correo electrónico, asegurando el uso de contraseñas seguras y autenticación multifactor (MFA) para prevenir accesos no autorizados.</td></tr>
<tr><td colspan="2">Método de prueba</td><td>Pruebas de funcionalidad y de seguridad</td></tr>
<tr><td rowspan="4">Tiempos de la prueba</td><td>Fecha de prueba</td><td>02/05/2026</td></tr>
<tr><td>Hora de inicio de prueba</td><td>9:10 am</td></tr>
<tr><td>Tiempo de prueba</td><td>3 horas</td></tr>
<tr><td>Disponibilidad de recursos para la prueba</td><td>Se garantiza que los recursos estarán disponibles según el día y horario establecido.</td></tr>
<tr><td>Los criterios utilizados para evaluar la prueba serán los siguientes:</td><td colspan="2">- Política de contraseña: El sistema debe exigir contraseñas que cumplan con los requisitos mínimos de seguridad (longitud, uso de mayúsculas, minúsculas, números y símbolos).
- Rechazo de credenciales débiles: El sistema debe impedir el acceso cuando se utiliza una contraseña que no cumpla con la política establecida.
- Autenticación Multifactor (MFA): La cuenta de correo debe tener MFA habilitado y activo.</td></tr>
</table>

	- **Acceso exitoso:** Solo se debe permitir el acceso al correo electrónico cuando el segundo factor sea ingresado correctamente. - **Registro de eventos:** El sistema debe registras los intentos de acceso fallidos y exitosos.	
Logística de la prueba	Responsable de la prueba	Jefe de Sistemas
	Integrantes involucrados	Jefe de Sistemas, Analista de sistemas
	Usuarios interesados	Jefe de Sistemas, jefe de Proyecto
	Responsables conformidad de la prueba	Jefe de Sistemas
Actividades de la Prueba	Actividad 1	Intentar acceder al servicio de correo con una contraseña débil para validar la política de contraseñas
	Actividad 2	Verificar que el sistema rechaza la contraseña débil y muestra el mensaje de error correspondiente
	Actividad 3	Verificar que está habilitada la autenticación multifactor (MFA) en la cuenta de correo
	Actividad 4	Intentar iniciar sesión sin el segundo factor y validar que el sistema lo solicita
	Actividad 5	Ingresar el código del segundo factor correctamente y verificar el acceso exitoso al correo electrónico
Nota: Las actividades serán ejecutadas en el entorno de producción de pruebas del servicio de correo electrónico (G-Suite / Google Workspace) de la organización.		

A1.1 Intentar acceder al servicio de correo con una contraseña débil para validar la política de contraseñas

Se realiza un intento de inicio de sesión en el servicio de correo electrónico utilizando una contraseña que no cumple con los requisitos mínimos de seguridad establecidos por la organización, tales como longitud insuficiente o ausencia de caracteres especiales. El propósito de esta actividad es verificar que el sistema identifique y bloquee el uso de credenciales inseguras.

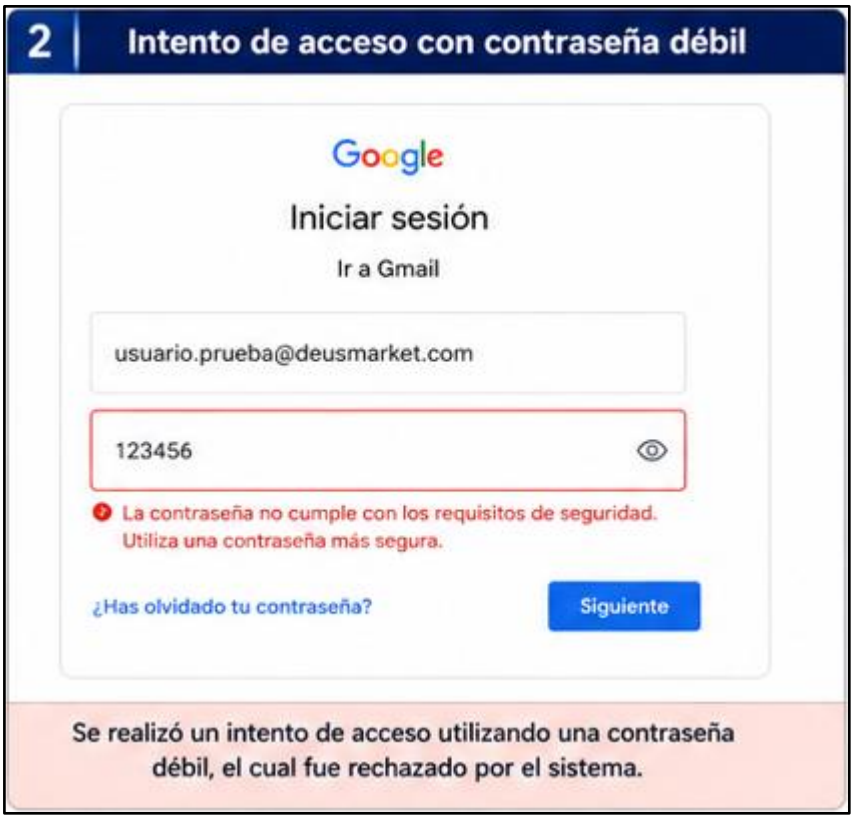
Tabla 14
Política de Contraseña Segura



A1.2 Verificar que el sistema rechaza la contraseña débil y muestra el mensaje de error correspondiente

Posterior al intento de acceso con una contraseña débil, se valida que el sistema rechace el inicio de sesión y muestre un mensaje de error indicando el incumplimiento de la política de contraseñas. Esta actividad permite comprobar la correcta aplicación de controles de seguridad en la autenticación.

Tabla 15
Intento de acceso con contraseña débil



A1.3 Verificar que está habilitada la autenticación multifactor (MFA) en la cuenta de correo.

Se revisa la configuración de la cuenta de correo electrónico para verificar que la autenticación multifactor (MFA) se encuentre habilitada y operativa. Esto incluye la validación de los métodos de verificación configurados, como aplicaciones autenticadoras o dispositivos móviles registrados.

Tabla 16
Activación de MFA



A1.4 Intentar iniciar sesión sin el segundo factor y validar que el sistema lo solicita.

Se intenta iniciar sesión sin completar el segundo factor de autenticación, con el fin de comprobar que el sistema solicite obligatoriamente un código de verificación adicional antes de permitir el acceso. Esta actividad valida la implementación del control de doble factor.

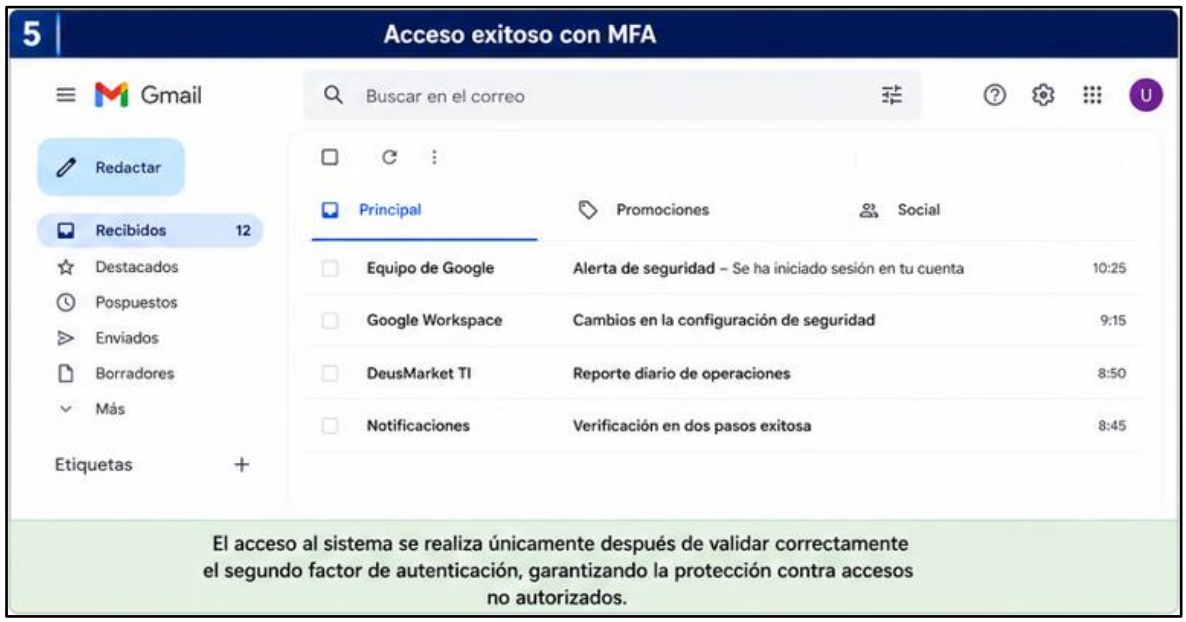
Tabla 177
Solicitud de Segundo Factor



A1.5 Ingresar el código del segundo factor correctamente y verificar el acceso exitoso al correo electrónico

Se ingresa correctamente el código de verificación generado por el segundo factor de autenticación y se valida que el acceso al servicio de correo electrónico se realice de manera exitosa. Esta actividad confirma que el sistema permite el acceso únicamente cuando se cumplen todos los mecanismos de seguridad establecidos.

Tabla 188
Acceso exitoso con MFA



Ejecución de prueba OE1

Inicialmente, el riesgo asociado al servicio de correo electrónico fue clasificado en un estado inaceptable, debido a la posibilidad de accesos no autorizados ocasionados por el uso de contraseñas débiles y la ausencia de mecanismos adicionales de autenticación. Esta situación incrementaba la probabilidad de compromiso de cuentas, afectando la confidencialidad e integridad de la información gestionada a través del servicio de correo electrónico.

Con el objetivo de mitigar este riesgo, se implementaron controles de seguridad orientados al fortalecimiento del proceso de autenticación. Entre las medidas adoptadas se encuentran la aplicación de políticas de contraseñas seguras, el bloqueo de credenciales que no cumplen los requisitos mínimos de complejidad, la habilitación de autenticación multifactor (MFA) y el registro de eventos relacionados con los intentos de acceso.

Posteriormente, mediante la ejecución de las pruebas definidas en el plan de pruebas CP-EMAIL-01, se verificó que los controles implementados funcionan correctamente, impidiendo el acceso mediante contraseñas inseguras y exigiendo la validación de un segundo factor de autenticación antes de permitir el ingreso al sistema.

Como resultado de la aplicación y validación de estos controles, la probabilidad de ocurrencia del riesgo se redujo significativamente, disminuyendo su nivel de criticidad y

permitiendo que el riesgo pase de un estado inaceptable a un estado aceptable, de acuerdo con los criterios establecidos en la matriz de riesgos de la organización.

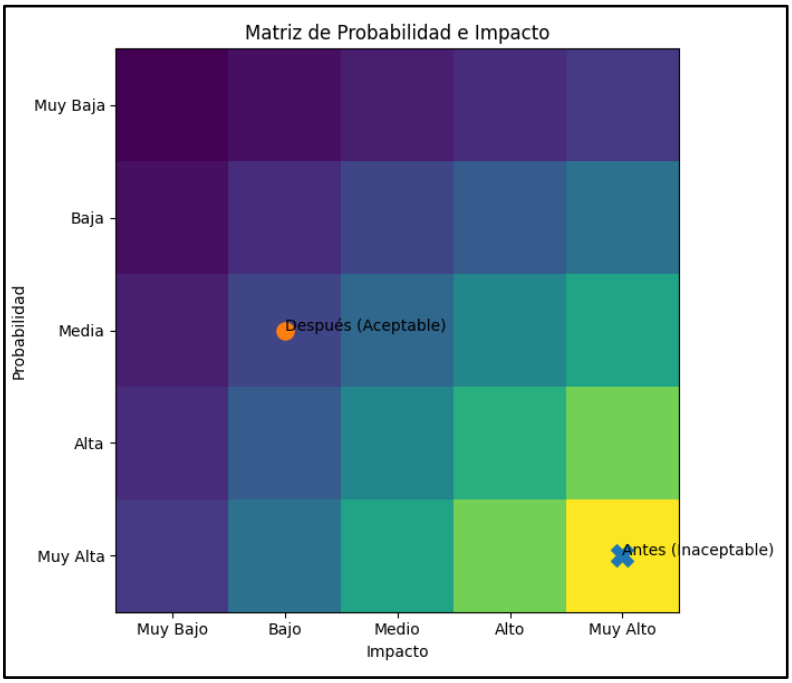
Tabla 19
Resultados de los datos de acuerdo con los criterios establecidos

ITEM	CRITERIO	VALOR ESPERADO	VALOR OBTENIDO
1	Política de contraseñas	3	5
2	Rechazo de credenciales débiles	4	5
3	Autenticación multifactor (MFA)	3	5
4	Solicitud de segundo factor	2	5
5	Acceso controlado	3	5

Nota: Tabla de criterios con % de criticidad del 100%.

VALOR	NIVEL DEL VALOR	DESCRIPCIÓN
1	Bajo	No se cuenta con ningún nivel de control o conocimiento
2	Regular	Se cuenta con nivel regular
3	Básico	Se cuenta con nivel Básico
4	Estándar	Se cuenta con nivel Estándar o Normal
5	Alto	Nivel Alto

Tabla 20
Matriz riesgo de Probabilidad e Impacto



Conclusión del Plan de Prueba OE1

De acuerdo con los resultados obtenidos, se determinó que los mecanismos de seguridad implementados en el servicio de correo electrónico cumplen satisfactoriamente con los criterios de evaluación establecidos. La validación de las políticas de contraseñas, el uso obligatorio de autenticación multifactor y el monitoreo de eventos de acceso fortalecen los controles de seguridad del sistema, disminuyendo la probabilidad de compromiso de cuentas y acceso indebido a la información. Por lo tanto, se concluye que las medidas implementadas contribuyen de manera efectiva a la mitigación del riesgo identificado, mejorando el nivel de seguridad del servicio de correo electrónico y garantizando una adecuada protección de los activos de información de la organización.

5.3.2 Elaboración de prueba Objetivo E2

Objetivo 2: Diseñar el sistema de criptografía asimétrica para el envío seguro de mensajes SMTP, especificando los algoritmos, la gestión de claves y los mecanismos de cifrado y firma digital, en un plazo de 4 semanas.

Tabla *21*1
Tabla Cuadro de preparación de Objetivo Estratégico2

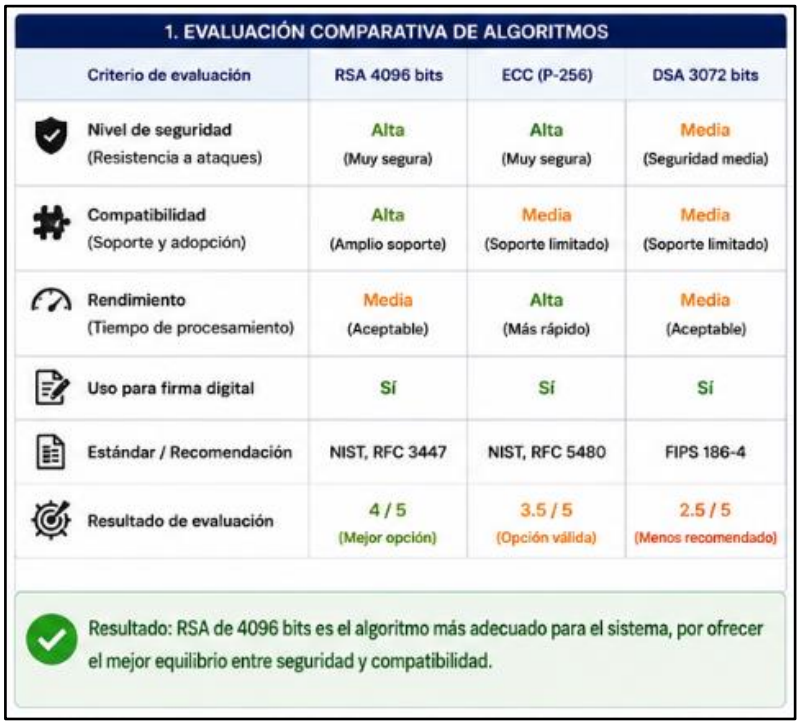
<table>
<tr><td colspan="3">Plan de Prueba</td></tr>
<tr><td colspan="3">Objetivo: Validar el diseño del sistema de criptografía asimétrica para el envío seguro de mensajes SMTP, verificando los mecanismos de cifrado, firma digital y gestión de claves para garantizar la confidencialidad, integridad y autenticidad de la información.</td></tr>
<tr><td colspan="2">Método de prueba</td><td>Evaluación de Mecanismos</td></tr>
<tr><td rowspan="4">Tiempos de la prueba</td><td>Fecha de prueba</td><td>03/05/2026</td></tr>
<tr><td>Hora de inicio de prueba</td><td>10:15 am</td></tr>
<tr><td>Tiempo de prueba</td><td>3 horas</td></tr>
<tr><td>Disponibilidad de recursos para la prueba</td><td>Se garantiza que los recursos estarán disponibles según el día y horario establecido.</td></tr>
<tr><td>Los criterios utilizados para evaluar la prueba serán los siguientes:</td><td colspan="2">- Se verificará la correcta definición de los mecanismos de criptografía asimétrica a utilizar.
- Se evaluará la selección de algoritmos criptográficos de acuerdo con los estándares conocidos.
- Se validará la definición de los mecanismos de generación y gestión de claves criptográficas.
- Se verificará la correcta implementación conceptual de los procesos de cifrado y descifrado.
- Se evaluará la incorporación de mecanismos de firma digital para garantizar autenticidad e integridad.
- Se validará la arquitectura propuesta para la integración del criptosistema en el servicio de SMTP.</td></tr>
<tr><td rowspan="4">Logística de la prueba</td><td>Responsable de la prueba</td><td>Jefe de Sistemas</td></tr>
<tr><td>Integrantes involucrados</td><td>Jefe de Sistemas, Analista de sistemas</td></tr>
<tr><td>Usuarios interesados</td><td>Jefe de Sistemas, jefe de Proyecto</td></tr>
<tr><td>Responsables conformidad de la prueba</td><td>Jefe de Sistemas</td></tr>
<tr><td rowspan="2">Actividades de la Prueba</td><td>Actividad 1</td><td>Realizar tabla comparativa de los algoritmos a utilizar</td></tr>
<tr><td>Actividad 2</td><td>Verificar el mecanismo de encriptación</td></tr>
</table>

	Actividad 3	Verificar el tipo de llaves de encriptación.
	Actividad 4	Verificar la firma digital
	Actividad 5	Validación del diseño propuestos de la arquitectura de integración

A2.1: Tabla comparativa de Algoritmos a utilizar

La Figura 55 presenta una evaluación comparativa de los algoritmos criptográficos RSA 4096 bits, ECC (P-256) y DSA 3072 bits, considerando criterios fundamentales para el diseño de un sistema de criptografía asimétrica orientado al envío seguro de mensajes SMTP. Los criterios evaluados incluyen el nivel de seguridad frente a ataques, la compatibilidad con herramientas y servicios existentes, el rendimiento en términos de tiempo de procesamiento, la capacidad de utilizar firmas digitales y el cumplimiento de estándares internacionales reconocidos.

Figura 55
Evaluación Comparativa de Algoritmos



La Figura 56 muestra los resultados de las pruebas realizadas a los algoritmos RSA 4096 bits, ECC (P-256) y DSA 3072 bits en un entorno controlado. Se verificó la correcta

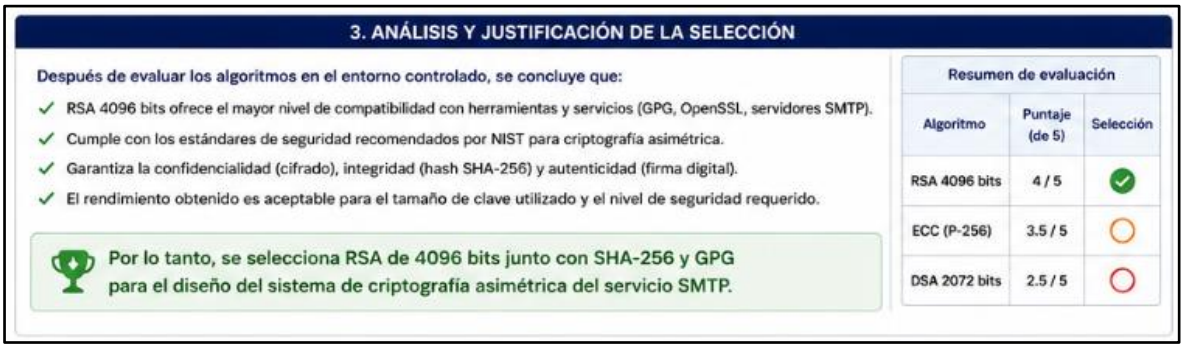
generación de pares de claves criptográficas mediante la herramienta GPG, obteniéndose claves públicas y privadas válidas para cada algoritmo. Asimismo, se realizaron pruebas de cifrado y descifrado utilizando mensajes de prueba, logrando recuperar correctamente la información original y mantener su integridad. Los resultados evidencian que los tres algoritmos funcionaron satisfactoriamente en las operaciones criptográficas evaluadas. Finalmente, aunque ECC presentó el mejor rendimiento, RSA 4096 bits fue considerado la alternativa más adecuada debido a su mayor nivel de seguridad y compatibilidad con la solución propuesta.

Figura 56
Pruebas de funcionamiento de algoritmos



La Figura 57 presenta el análisis y la justificación de la selección del algoritmo criptográfico para el sistema propuesto. Los resultados muestran que RSA 4096 bits obtuvo la mayor puntuación de evaluación debido a su alto nivel de seguridad, amplia compatibilidad y cumplimiento de estándares internacionales como NIST. Asimismo, garantiza la confidencialidad mediante cifrado, la integridad mediante SHA-256 y la autenticidad mediante firma digital. Aunque ECC y DSA son alternativas viables, obtuvieron menores puntuaciones en la evaluación general. Por ello, se seleccionó RSA 4096 bits junto con SHA-256 y GPG como la solución criptográfica para el envío seguro de mensajes SMTP.

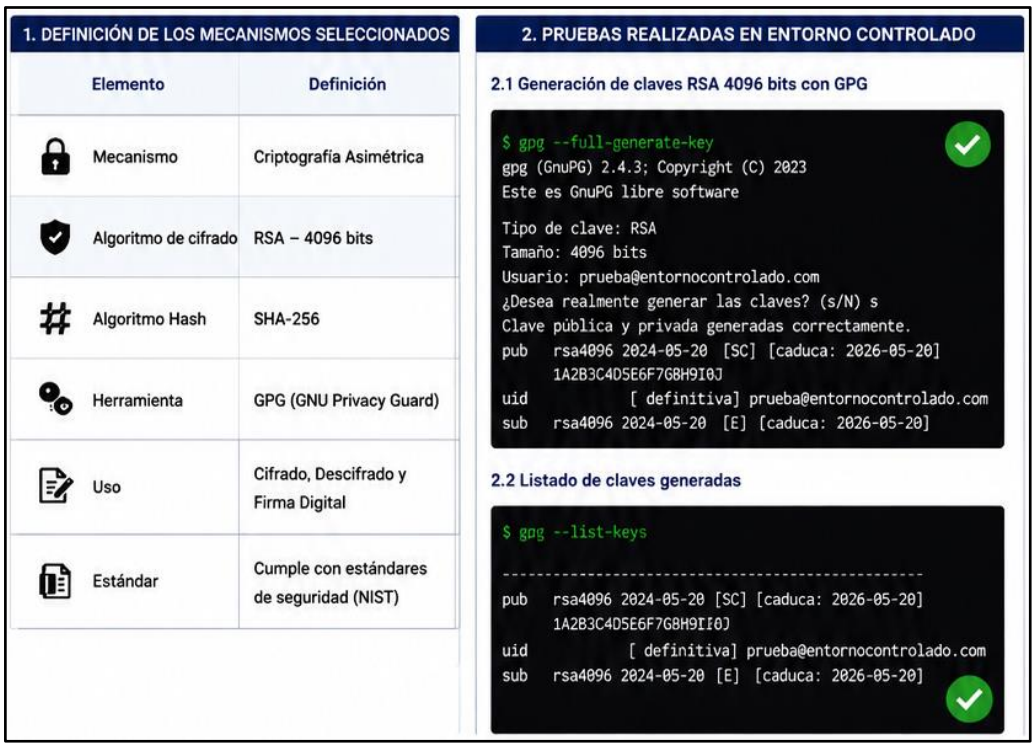
Figura 57
Evaluación Comparativa de Algoritmos



A2.2: Verificar el mecanismo de encriptación

Se verificará el correcto funcionamiento del mecanismo de encriptación mediante pruebas de cifrado y descifrado realizadas en un entorno controlado. Para ello, se evaluará la capacidad del sistema para proteger la información durante su transmisión y recuperar posteriormente el contenido original utilizando las claves criptográficas correspondientes. Los resultados permitirán determinar si el mecanismo garantiza la confidencialidad de los datos procesados.

Figura 588
Definición de los mecanismos y pruebas



La Figura 58 muestra los mecanismos criptográficos seleccionados para la solución propuesta y las pruebas realizadas en un entorno controlado. Se observa el uso de criptografía asimétrica basada en RSA 4096 bits, SHA-256 y la herramienta GPG. Asimismo, se evidencia la generación correcta del par de claves criptográficas y su posterior verificación. Los resultados obtenidos confirman la viabilidad de los mecanismos seleccionados para proteger la información intercambiada mediante SMTP.

Figura 599
Prueba de funcionamiento de Mecanismos



La Figura 59 presenta la prueba de funcionamiento del mecanismo de criptografía asimétrica en un entorno controlado. Se observa el proceso completo de cifrado del mensaje mediante la clave pública del destinatario, seguido del descifrado utilizando la clave privada correspondiente. Asimismo, se realizó una verificación de integridad comparando el mensaje original con el mensaje recuperado. Los resultados evidencian que la información fue recuperada sin modificaciones, confirmando el correcto funcionamiento del mecanismo criptográfico.

A2.3: Verificar el tipo de llaves de encriptación

Se verificará el tipo de llaves de encriptación utilizadas en el sistema, evaluando sus características, longitud y nivel de seguridad conforme a estándares criptográficos reconocidos. Asimismo, se comprobará que las claves seleccionadas sean adecuadas para

los procesos de cifrado, descifrado y firma digital propuestos. Los resultados permitirán determinar si las llaves criptográficas garantizan un nivel de protección acorde con los requisitos de seguridad establecidos.

Figura 600
Proceso de gestión de claves y pruebas



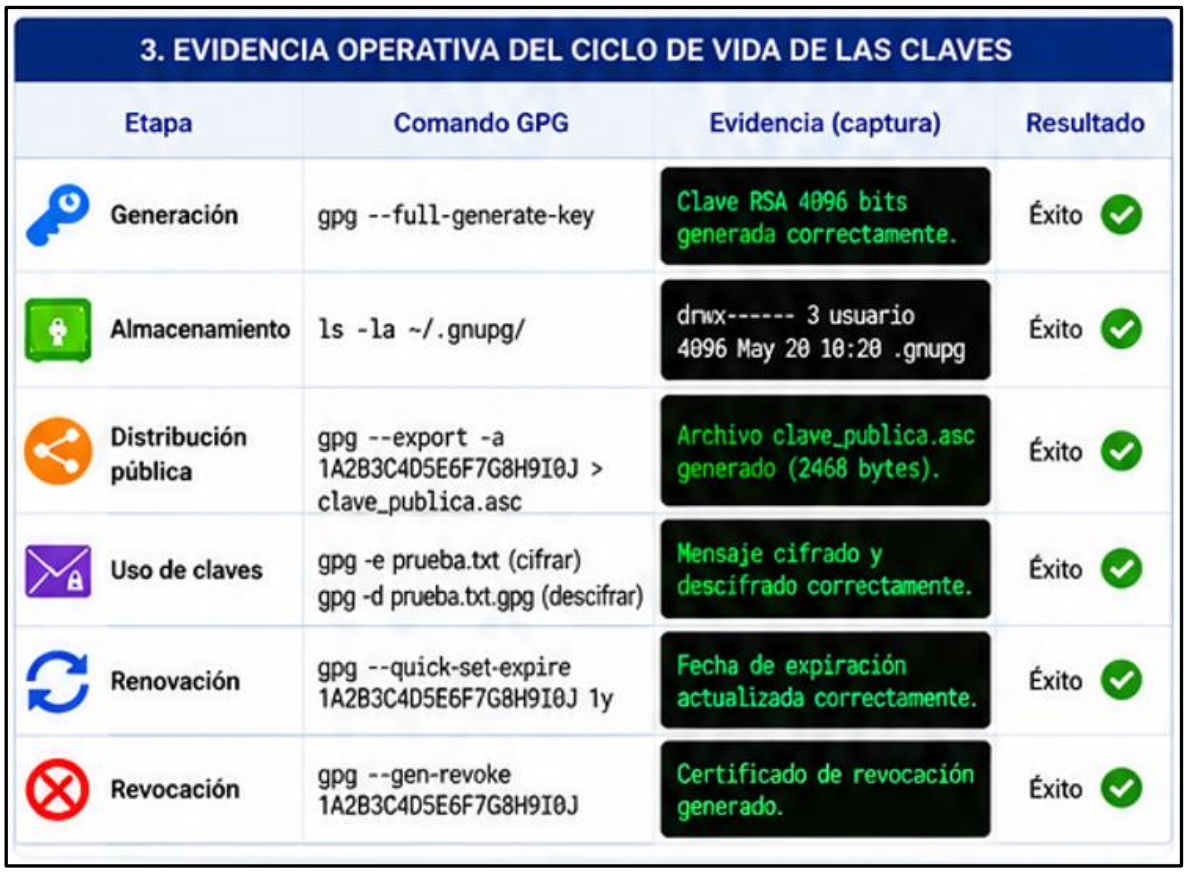
La Figura 60 muestra el proceso de gestión de claves criptográficas propuesto para el sistema. El ciclo comprende la generación, almacenamiento seguro, distribución, uso, renovación y revocación de las claves criptográficas. Cada etapa contribuye a garantizar la protección y administración adecuada de las claves durante todo su ciclo de vida. Asimismo, el proceso permite fortalecer la seguridad de las comunicaciones mediante el uso controlado de las claves.

Figura 611
Pruebas realizas en GPG



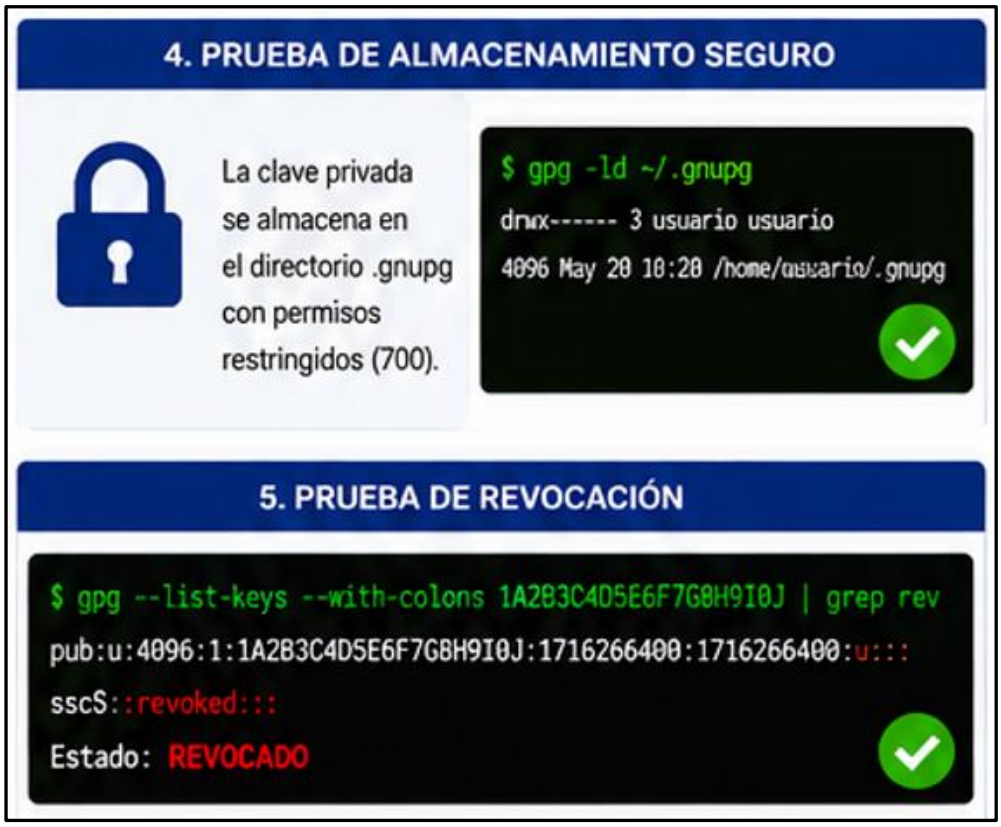
La Figura 61 presenta las pruebas realizadas para validar el proceso de gestión de claves criptográficas utilizando GPG en un entorno controlado. Se evidencia la generación del par de claves RSA de 4096 bits, la verificación de las claves creadas y la exportación de la clave pública para su distribución. Los resultados muestran que todas las operaciones se ejecutaron correctamente, confirmando la viabilidad del proceso de gestión de claves propuesto.

Figura 622
Evidencia Operativa de ciclo de vida de las claves



La Figura 62 presenta la evidencia operativa del ciclo de vida de las claves criptográficas gestionadas mediante GPG. Se muestran las actividades de generación, almacenamiento, distribución, uso, renovación y revocación de claves, junto con los resultados obtenidos en cada etapa. Los resultados evidencian que todas las operaciones fueron ejecutadas correctamente, garantizando una gestión segura y controlada de las claves criptográficas durante su ciclo de vida.

Figura 633
Prueba de Almacenamiento seguro



La Figura 63 presenta las pruebas de almacenamiento seguro y revocación de claves criptográficas realizadas en un entorno controlado. Se verifica que la clave privada se almacena en un directorio protegido con permisos restringidos, garantizando su confidencialidad. Asimismo, se evidencia la correcta ejecución del proceso de revocación de claves, permitiendo invalidar una clave comprometida cuando sea necesario. Los resultados confirman la efectividad de los mecanismos de protección y control implementados para la gestión segura de claves criptográficas.

A2.4: Verificar firma digital

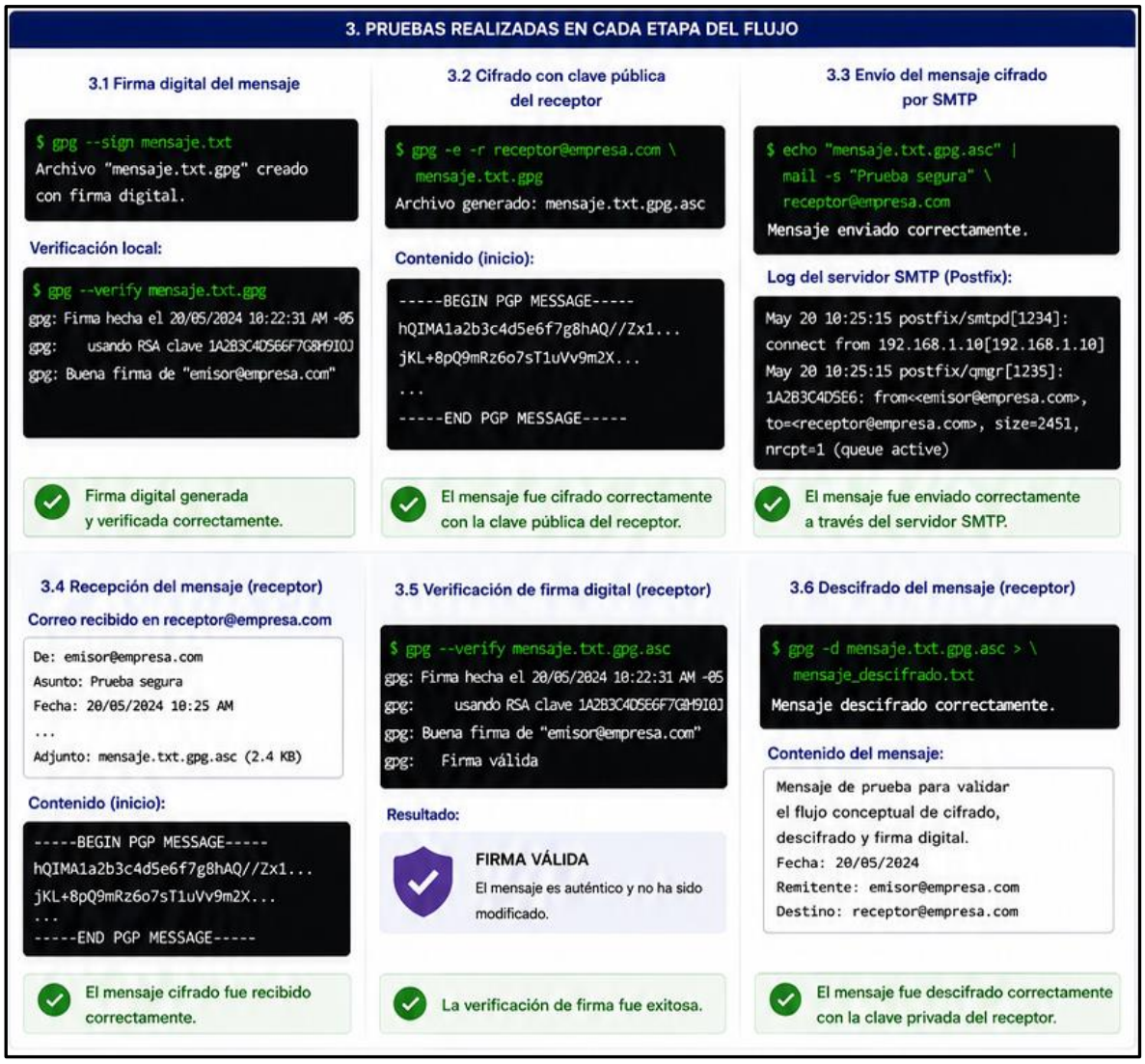
Se verificará el funcionamiento del mecanismo de firma digital mediante pruebas realizadas en un entorno controlado. Para ello, se evaluará la capacidad del sistema para garantizar la autenticidad del remitente y la integridad de la información transmitida, validando que cualquier modificación no autorizada del mensaje pueda ser detectada. Los resultados permitirán determinar la efectividad de la firma digital como mecanismo de seguridad dentro del sistema propuesto.

Figura 644
Verificar firma digital y prueba



La Figura 64 presenta el flujo conceptual propuesto para el envío seguro de mensajes SMTP mediante criptografía asimétrica. El proceso comprende la firma digital, el cifrado del mensaje, la transmisión a través del servidor SMTP, la verificación de la firma y el descifrado por parte del receptor. Asimismo, se muestran las características del entorno de prueba utilizado para la validación de la solución. Este flujo garantiza las propiedades de confidencialidad, integridad y autenticidad de la información intercambiada.

Figura 655
Pruebas realizadas en cada etapa del flujo



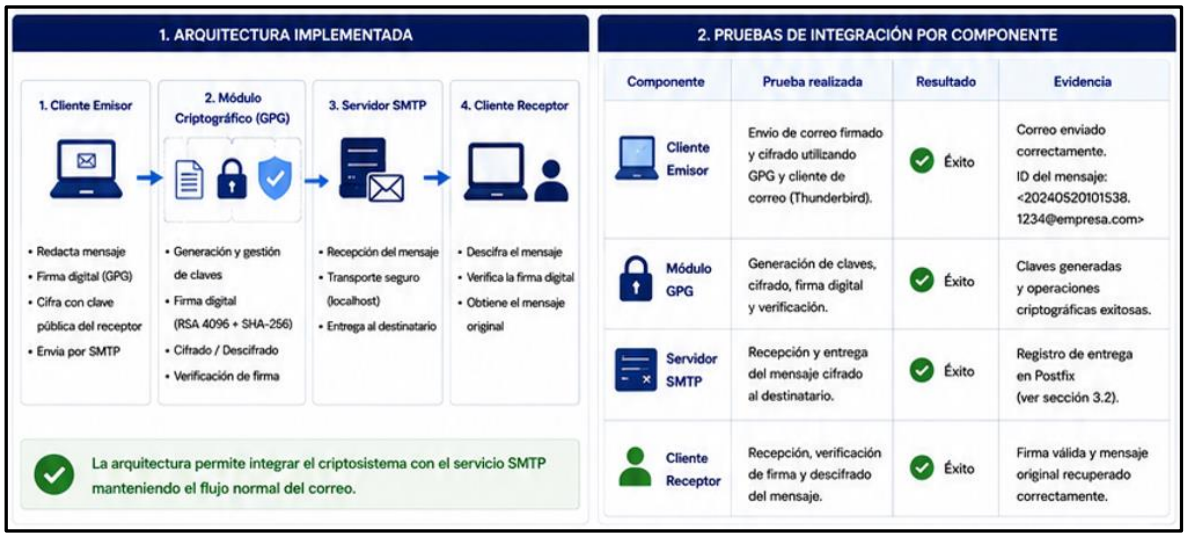
La Figura 65 presenta las pruebas realizadas en cada etapa del flujo de criptografía asimétrica para el envío seguro de mensajes SMTP. Se evidencia la ejecución correcta de los procesos de firma digital, cifrado, envío, recepción, verificación de firma y descifrado del mensaje. Los resultados obtenidos demuestran que la información fue transmitida y recuperada exitosamente sin alteraciones. Asimismo, se verificó la autenticidad del remitente y la integridad del mensaje mediante el uso de firma digital.

A2.5: Validación del diseño propuestos de la arquitectura de integración

Se validará el diseño propuesto de la arquitectura de integración del criptosistema con el servicio SMTP, verificando la correcta interacción entre los componentes involucrados en el proceso de envío y recepción de mensajes. Asimismo, se evaluará que la arquitectura

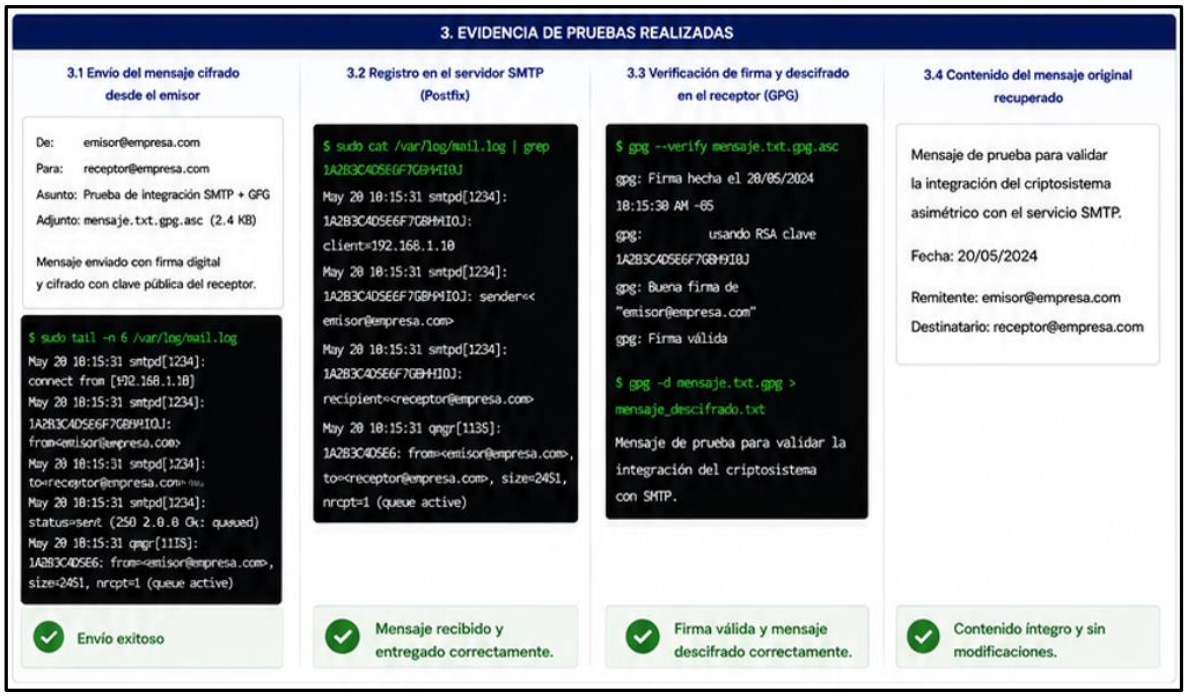
permita incorporar mecanismos de cifrado, gestión de claves y firma digital sin afectar la operatividad del sistema. Los resultados permitirán determinar si la solución propuesta cumple con los requisitos de confidencialidad, integridad y autenticidad establecidos para la transmisión segura de información.

Figura 666
Validación de la arquitectura criptográfica propuesta



La Figura 66 presenta la arquitectura propuesta para la integración del criptosistema con el servicio SMTP y los resultados de las pruebas realizadas a cada componente. Se observa la interacción entre el cliente emisor, el módulo criptográfico GPG, el servidor SMTP y el cliente receptor durante el proceso de envío seguro de mensajes. Asimismo, las pruebas de integración evidencian que cada componente ejecutó correctamente sus funciones de cifrado, firma digital, transporte y verificación. Los resultados obtenidos confirman la viabilidad de la arquitectura propuesta para garantizar comunicaciones seguras mediante SMTP.

Figura 677
Evidencia de Pruebas realizadas



La Figura 67 presenta la evidencia de las pruebas realizadas para validar la integración del criptosistema con el servicio SMTP. Se observa el envío exitoso de un mensaje cifrado, su registro en el servidor SMTP, la verificación de la firma digital y el proceso de descifrado en el receptor. Asimismo, se confirma que el contenido original fue recuperado íntegramente y sin modificaciones. Los resultados demuestran el correcto funcionamiento de la solución propuesta durante todo el proceso de comunicación segura.

Ejecución de prueba OE2

De acuerdo con los resultados obtenidos, se concluye que la arquitectura criptográfica diseñada es técnicamente viable y cumple con los requisitos de seguridad planteados para el proyecto. La integración de mecanismos de criptografía asimétrica, gestión de claves y firma digital permite establecer un entorno seguro para el intercambio de mensajes SMTP, reduciendo significativamente los riesgos asociados a la exposición, modificación o suplantación de información durante la transmisión.

Tabla 222
Resultados de la Validación de la arquitectura

N°	Criterio evaluado	Resultado esperado	Resultado obtenido (1 a 5)		Cumplimiento
			Valor (1–5)	Justificación	
1	Cumplimiento de requisitos de seguridad	Todos los requisitos de seguridad definidos por la arquitectura deben estar cubiertos.	5	La arquitectura cubre confidencialidad, integridad, autenticidad, disponibilidad y gestión de claves.	CUMPLE
2	Integración de componentes	Los componentes deben interactuar correctamente entre sí.	5	Remitente, gestión de claves, servidor SMTP y destinatario interactúan de forma correcta.	CUMPLE
3	Flujo de procesos	El flujo de cifrado, firma, envío, verificación y descifrado debe ser coherente y seguro.	5	El flujo definido es coherente, secuencial y asegura la protección de la información.	CUMPLE
4	Gestión de claves	La gestión de claves debe ser adecuada, segura y separada por roles.	5	Se definen pares de claves por rol (remitente y destinatario) con separación adecuada.	CUMPLE
5	Disponibilidad del servidor SMTP	El servidor SMTP debe garantizar disponibilidad y transmisión segura (TLS/SSL).	5	La conexión segura TLS/SSL y la disponibilidad del servidor aseguran la entrega del mensaje.	CUMPLE
6	Arquitectura coherente e integrada	La arquitectura debe ser coherente, modular y alineada a los objetivos del sistema.	5	La arquitectura es modular, coherente y cumple con los objetivos del diseño propuesto.	CUMPLE

PUNTAJE TOTAL	PROMEDIO	RESULTADO GLOBAL
30 / 30	5.00 / 5	ARQUITECTURA VALIDADA
(Máximo: 30 puntos)	(Escala: 1 a 5)	La arquitectura propuesta es segura, confiable y cumple con todos los requisitos definidos.

Tabla 233
Tabla de niveles de Valores

VALOR	NIVEL DEL VALOR	DESCRIPCIÓN
1	Bajo	No se cuenta con ningún nivel de control
2	Regular	Se cuenta con nivel regular
3	Básico	Se cuenta con nivel Básico
4	Estándar	Se cuenta con nivel Estándar o Normal
5	Alto	Nivel Alto

Conclusión del Plan de Prueba OE2

Los resultados obtenidos durante la ejecución del plan de pruebas evidencian que el diseño del sistema de criptografía asimétrica cumple con los requisitos funcionales y de seguridad establecidos para el proyecto. La validación de la arquitectura propuesta permitió comprobar la correcta integración de los mecanismos de gestión de claves, cifrado y firma digital,

asegurando la protección de la información durante su transmisión. Por lo tanto, se concluye que la solución diseñada constituye una alternativa viable para fortalecer la seguridad del servicio SMTP y mitigar los riesgos asociados al acceso no autorizado, alteración o exposición de los mensajes electrónicos.

5.3.3 Elaboración de prueba Objetivo E3

Objetivo 3: Diseñar el modelo funcional del sistema, incluyendo los procesos de codificación, envío y almacenamiento de mensajes, mediante diagramas y especificaciones técnicas, en un plazo de 4 semanas.

Tabla 24
Tabla Cuadro de preparación de Objetivo Estratégico 3

Plan de Prueba		
Objetivo: Validar el funcionamiento del mecanismo de criptografía asimétrica mediante el cifrado, descifrado y verificación de archivos utilizando GNU Privacy Guard (GPG), garantizando la confidencialidad e integridad de la información procesada.		
Método de prueba		Evaluación de Mecanismos
Tiempos de la prueba	Fecha de prueba	04/05/2026
	Hora de inicio de prueba	10:15 am
	Tiempo de prueba	3 horas
	Disponibilidad de recursos para la prueba	Se garantiza que los recursos estarán disponibles según el día y horario establecido.
Los criterios utilizados para evaluar la prueba serán los siguientes:		- Se realizará la encriptación de datos utilizando GPG de diferentes archivos, se validará el cifrado y descifrado. - Se generará un informe que detallará los resultados obtenidos
Logística de la prueba	Responsable de la prueba	Jefe de Sistemas
	Integrantes involucrados	Jefe de Sistemas, Analista de sistemas
	Usuarios interesados	Jefe de Sistemas, jefe de Proyecto
	Responsables conformidad de la prueba	Jefe de Sistemas
Actividades de la Prueba	Actividad 1	Creación de archivos para realizar encriptación
	Actividad 2	Cifrado de Archivos
	Actividad 3	Descifrado de Archivos
	Actividad 4	Verificar Archivos
Herramientas		Lenguaje de Programación Python, herramienta de cifrado GPG

Ejecución de prueba OE 3

Se propuso implementar mecanismos de encriptación basados en el estándar GPG para cifrar datos, considerando archivos de texto y documentos.

A3.1 Creación de archivos para realizar la encriptación

Realizamos la creación de archivos de prueba para poder validar el cumplimiento del Objetivo Especifico 3

```
1. echo " Este es el primer archivo de prueba" > test1.txt
2. echo "print(' Este archivo de prueba es una secuencia de comandos de Python ')" > test2.py
3. echo " Este es el último archivo de prueba" > test3.txt
```

Ejecutamos un cat para poder ver el contenido del archivo test1.

```
1. cat test1.txt
```

```
1. Output
2. Este es el primer archivo de prueba
```

A.3.2 Cifrado de Archivos

Realizamos la ejecución del código para la firma y encriptación de los archivos

```
proyecto@deusmarket:~/python-prueba$ firmados
EL directorio firmas fue creado
test2.py  signature created
test1.txt  signature created
test3.txt  signature created
```

Como se puede apreciar tenemos un archivo llamado test1 firmado.

```
proyecto@deusmarket:~/python-prueba$ cat firmas/test1.txt.sig
-----BEGIN PGP SIGNATURE-----

iQGzBAABCgAdFiEEzRWh5ve/4qtvMQsP4LVshuBTlMEFAmRYJKgACgkQ4LVshuBT
lMGyjgv+P0xKD/lvPrSCaRUmhSmfgnQzop0PTeJ6/DIIyiLFZM9OB262dAm1OQGs
fTSDWl4eCVXq/KYg2VBPNccuP0NcqndtrpIz/QMMZWE2yARiYFI9IFqiPd+YIVuY
94OKL4cUJa8mH4SUmdafCOGF8yahHTkX/jS8I18oRTC1yptkxE3PkOZgyCdMTeWi
HAgoUMOUwntU3yrRhX20olpJBICqaj+U4PI+QsZ+l5xL9YmkTRss1KiHZaSN+ZHv
YB+lVfDP/APfoocmVEuCVMJ4slqRBh+1wzvYViaS9WPr4xX/MkVWFB84gY3tnYWJ
HwXmI7EsqKsDzrbZoJKAM3+ZrDrwvyq9ZxIqbUn1G66IU+GmQXHp6K97UxYCHhyh
to1IspXEeJP/SfnWcY4Dvt2rDfnV5OdUJH7zCO5TKsT7xElqod/WsYxOVTEUk9k0
GfzgZ7lKcDgPIXiiWnxDN+As2nMfAjPU33r3XD3ysvbRV3nxaqX7g6fmSSfl3iTZ
cVdiwlwD
=wk8F
-----END PGP SIGNATURE-----
```

Posteriormente realizaremos la encriptación lo cual nos da el siguiente resultado. Se puede apreciar que la encriptación de datos fue correcta.

```
proyecto@deusmarket:~/python-prueba$ encriptados
Se creo el directorio encriptados
ok:  True
status:  encryption ok
stderr:  [GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] ENCRYPTION_COMPLIANCE_MODE 23
[GNUPG:] BEGIN_ENCRYPTION 2 9
[GNUPG:] END_ENCRYPTION

ok:  True
status:  encryption ok
stderr:  [GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] ENCRYPTION_COMPLIANCE_MODE 23
[GNUPG:] BEGIN_ENCRYPTION 2 9
[GNUPG:] END_ENCRYPTION

ok:  True
status:  encryption ok
stderr:  [GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] ENCRYPTION_COMPLIANCE_MODE 23
[GNUPG:] BEGIN_ENCRYPTION 2 9
[GNUPG:] END_ENCRYPTION
```

Y para realizar la desencriptación realizamos la ejecución del código, el cual nos brindara el siguiente resultado.

A.3.3 Descifrado de Archivos

```
proyecto@deusmarket:~/python-prueba/encriptados$ desencriptados
El directorio desencriptados fue creado
ok:  True
status:  decryption ok
stderr:  [GNUPG:] ENC_TO 3A9E3B52EE884B5C 1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
[GNUPG:] DECRYPTION_KEY D9FDCA3B68CD01F93FB4B9FD3A9E3B52EE884B5C CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 u
[GNUPG:] KEY_CONSIDERED CD15A1E6F7BFE2AB6F310B0FE0B56C86E05394C1 0
gpg: encrypted with 3072-bit RSA key, ID 3A9E3B52EE884B5C, created 2023-05-07
      "Deusmarket <ventasonline@deusmarket.pe>"
[GNUPG:] BEGIN_DECRYPTION
[GNUPG:] DECRYPTION_COMPLIANCE_MODE 23
[GNUPG:] DECRYPTION_INFO 2 9 0
[GNUPG:] PLAINTEXT 62 1683498651
[GNUPG:] PLAINTEXT_LENGTH 37
[GNUPG:] DECRYPTION_OKAY
[GNUPG:] GOODMDC
[GNUPG:] END_DECRYPTION
```

A.3.4 Verificar Archivos

Realizamos el cat para poder ver el contenido del archivo test1, y nos muestra el texto original del archivo.



Conclusión del Plan de Prueba OE3

De acuerdo con los resultados obtenidos, se determinó que los procesos definidos en el modelo funcional del sistema operan correctamente. La ejecución de las actividades de cifrado, descifrado y verificación de archivos permitió validar la efectividad de los mecanismos criptográficos implementados mediante GPG, garantizando la confidencialidad e integridad de la información procesada. Por lo tanto, se concluye que el modelo funcional propuesto es técnicamente viable y cumple con los objetivos establecidos para la protección de la información dentro de la solución planteada.

5.5.4 Elaboración de prueba Objetivo E4

Objetivo 4: Diseñar e implementar un sistema automatizado orientado a la codificación por bloques y la transferencia de información mediante AS2, haciendo uso del lenguaje Python, definiendo su arquitectura lógica, flujo de procesamiento y mecanismos de comunicación segura, en un plazo de 4 semanas.

Elaboración de prueba OE4

Tabla 255
Tabla Cuadro de preparación de Objetivo Estratégico 4

<table>
<tr><td colspan="3">Plan de Prueba</td></tr>
<tr><td colspan="3">Objetivo: Validar el envío y recepción segura de mensajes cifrados con proveedores externos, verificando el correcto funcionamiento de los mecanismos de criptografía implementados para garantizar la confidencialidad e integridad de la información intercambiada.</td></tr>
<tr><td colspan="2">Método de prueba</td><td>Prueba con Proveedor</td></tr>
<tr><td rowspan="4">Tiempos de la prueba</td><td>Fecha de prueba</td><td>06/05/2026</td></tr>
<tr><td>Hora de inicio de prueba</td><td>10:00 am</td></tr>
<tr><td>Tiempo de prueba</td><td>2 horas</td></tr>
<tr><td>Disponibilidad de recursos para la prueba</td><td>Se garantiza que los recursos estarán disponibles según el día y horario establecido.</td></tr>
<tr><td colspan="2">Los criterios utilizados para evaluar la prueba serán los siguientes:</td><td>- Se realizará la evaluación del área de compras para el envío y recepción de mensajes con los proveedores
- Se generará un informe que detallará los resultados obtenidos</td></tr>
<tr><td rowspan="4">Logística de la prueba</td><td>Responsable de la prueba</td><td>Jefe de Sistemas</td></tr>
<tr><td>Integrantes involucrados</td><td>Jefe de Sistemas, Analista de sistemas</td></tr>
<tr><td>Usuarios interesados</td><td>Jefe de Sistemas, jefe de Proyecto, jefe de Compras</td></tr>
<tr><td>Responsables conformidad de la prueba</td><td>Jefe de Sistemas</td></tr>
<tr><td>Actividades de la Prueba</td><td>Actividad 1</td><td>Prueba de envío y recepción de mensajes cifrados</td></tr>
</table>

Ejecución de prueba OE4

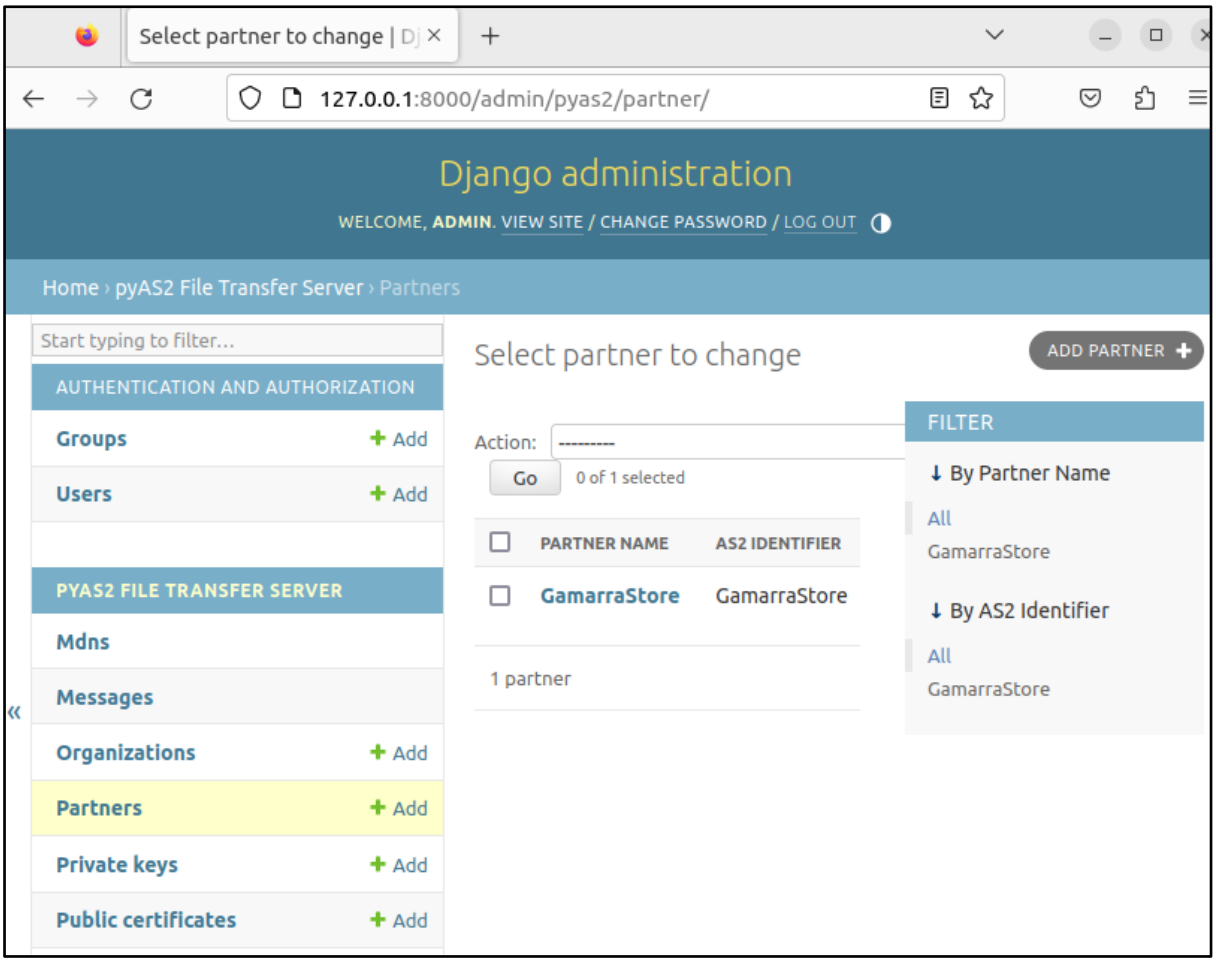
Se desarrollo un sistema para la codificación por lotes de archivos para ser enviados del servidor DeusMarket hacia el cliente en este caso GamarraStore, se realiza la configuración de los servidores en Python utilizando AS2. Adicionalmente se realizó la creación de llaves para la interacción entre los 2 servidores y a continuación se presenta él envió de mensajes cifrados.

A.4.1 Prueba de envío y recepción de mensajes cifrados

Envió de Mensajes Cifrados utilizando AS2 desde nuestro Servidor Principal DeusMarket

Seleccionaremos a nuestro Partner para él envió de mensaje, para el caso sería GamarraStore.

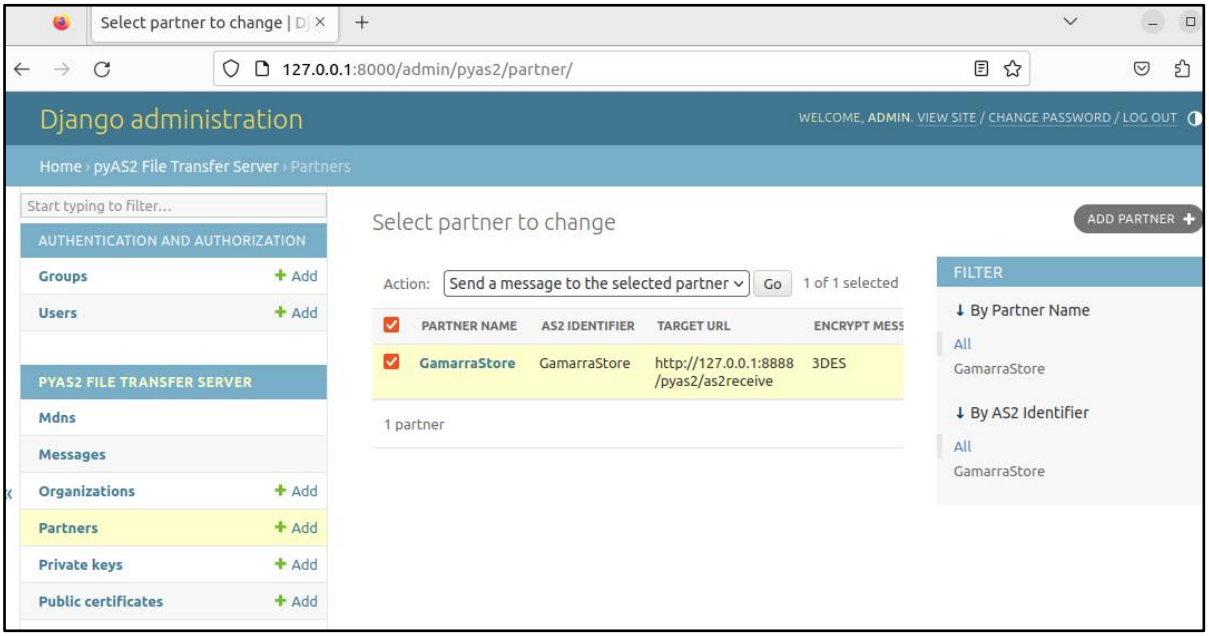
Figura 688
Evidencia de envío y recepción de mensajes cifrados



Seleccionamos a GamarraStore y en la acción elegimos:

Send a message to the selected partner.

Figura 699
Evidencia de selección de partner



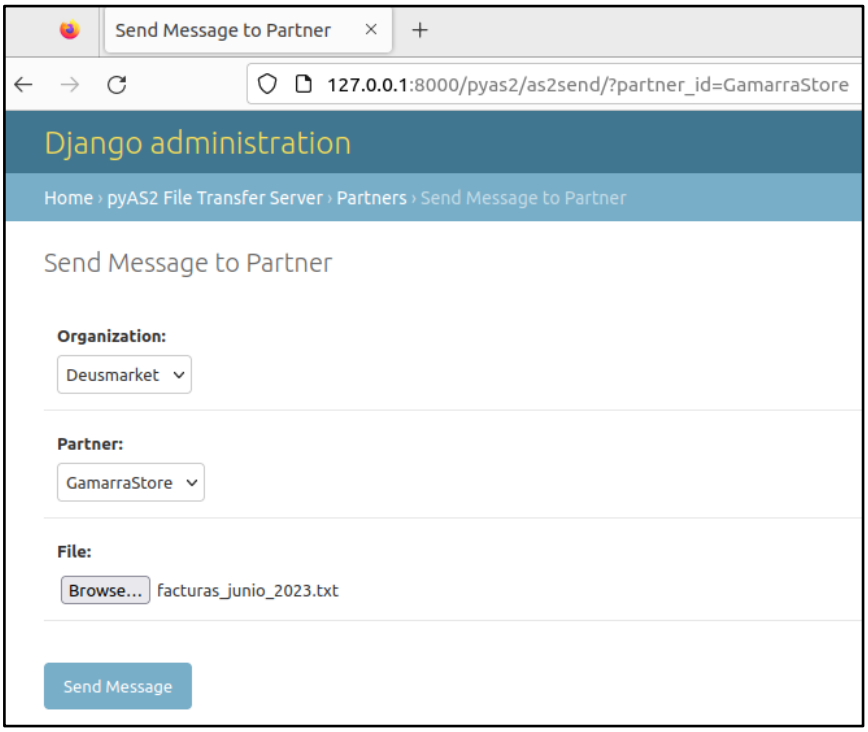
Ahora elegiremos lo siguiente para el envío de mensaje y archivos:

Organization: DeusMarket

Partner: GamarraStore

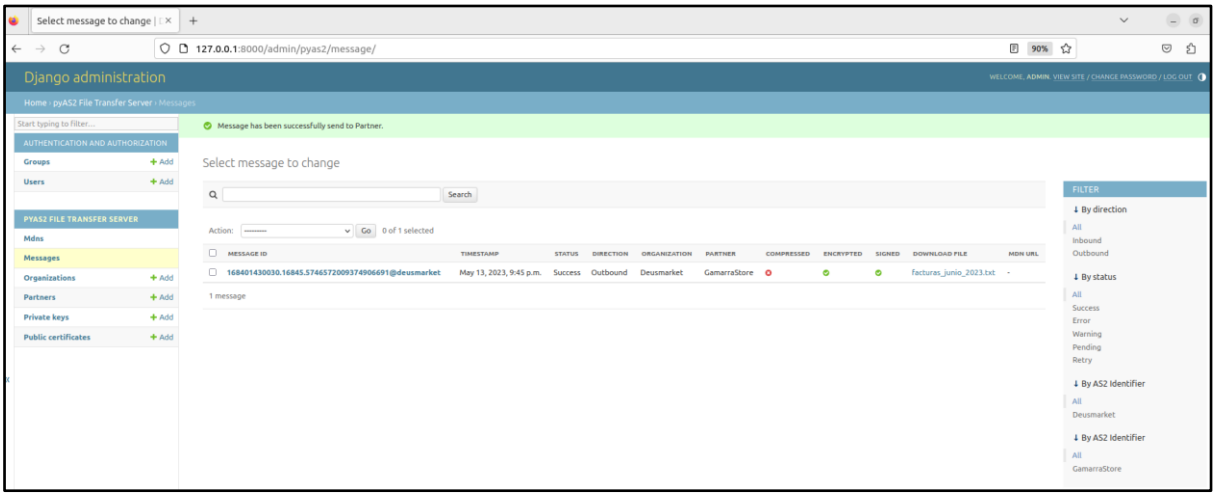
File: facturas_junio_2026

Figura 70
Evidencia de selección para el envío de mensajes y archivos



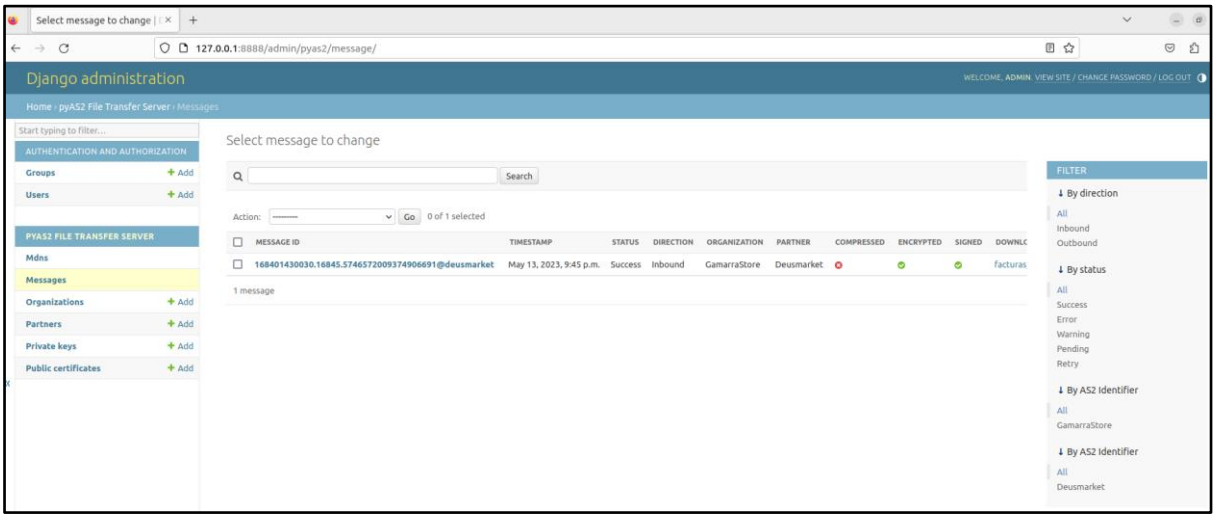
Apreciamos la confirmación del envió del mensaje realizado por DeusMarket

Figura 71
Confirmación del envío del mensaje



Podemos ver que el archivo envío por mensaje fue recibido por GamarraStore

Figura 72
Evidencia del envío del archivo a GamarraStore



Conclusión del Plan de Prueba OE4

De acuerdo con los resultados obtenidos, se determinó que el sistema automatizado de transferencia de información implementado cumple adecuadamente con los requisitos funcionales y de seguridad definidos para el proyecto. La validación del envío y recepción de mensajes cifrados permitió comprobar la correcta integración de los mecanismos criptográficos utilizados para la protección de la información durante su transmisión. Asimismo, se verificó la capacidad del sistema para garantizar la confidencialidad e integridad de los datos intercambiados entre las entidades participantes. Por lo tanto, se concluye que la solución desarrollada es técnicamente viable y contribuye significativamente al fortalecimiento de la seguridad en los procesos de intercambio electrónico de información.

6. CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusión

El análisis realizado evidenció que el proceso de envío y almacenamiento de mensajes SMTP presenta vulnerabilidades asociadas principalmente a la exposición de la información durante su transmisión y al acceso no autorizado a los datos, lo que puede comprometer la confidencialidad, integridad y autenticidad de la información intercambiada.

Los resultados obtenidos mediante las pruebas realizadas utilizando GPG (GNU Privacy Guard), Postfix SMTP, VirtualBox y Ubuntu 22.04 LTS evidenciaron que la solución criptográfica basada en RSA 4096 bits y SHA-256 garantiza la confidencialidad, integridad y autenticidad de los mensajes electrónicos. Asimismo, las pruebas de generación y gestión de claves, cifrado, descifrado y firma digital permitieron comprobar la correcta interacción entre los componentes del sistema. Finalmente, se determinó que la arquitectura propuesta constituye una alternativa técnicamente viable para fortalecer la seguridad del servicio SMTP y reducir los riesgos asociados a la exposición y modificación no autorizada de la información.

La ejecución de las pruebas mediante Python y la herramienta GPG permitió verificar el comportamiento del modelo funcional propuesto durante las operaciones de procesamiento de información. Las evidencias obtenidas demostraron que las actividades de codificación, cifrado, descifrado y validación de archivos se desarrollan de manera consistente, conservando la integridad del contenido procesado. En ese sentido, se comprobó que el modelo funcional definido proporciona una base adecuada para el tratamiento seguro de la información dentro de la solución planteada.

Las pruebas realizadas utilizando Python, el protocolo AS2 y la plataforma pyAS2 permitieron evidenciar el correcto funcionamiento del sistema automatizado de intercambio de información. Los resultados obtenidos durante el envío y recepción de mensajes cifrados demostraron la adecuada comunicación entre las entidades participantes y la efectividad de los mecanismos de protección implementados. En consecuencia, se comprobó que la solución propuesta constituye una alternativa confiable para garantizar la seguridad en la transferencia electrónica de información.

6.2 Recomendaciones

Establecer una Infraestructura de Clave Pública (PKI) que permita administrar de forma centralizada la emisión, distribución, renovación y revocación de certificados digitales. Esta mejora contribuirá a optimizar la gestión de claves criptográficas utilizadas en la organización y fortalecerá los mecanismos de confianza, autenticidad e integridad en las comunicaciones electrónicas.

Ampliar la implementación de la solución propuesta hacia otros procesos de intercambio electrónico de información, integrando protocolos y tecnologías complementarias como SFTP, API REST seguras o plataformas EDI. De esta manera, la organización podrá extender los beneficios de la automatización y la seguridad implementada a diferentes entornos de comunicación con clientes, proveedores y socios comerciales.

Desarrollar futuras investigaciones orientadas a la incorporación de tecnologías emergentes, tales como Blockchain para garantizar la trazabilidad y el no repudio de las transacciones, así como Inteligencia Artificial para la detección temprana de anomalías y amenazas en las comunicaciones electrónicas. Estas iniciativas permitirán incrementar los niveles de seguridad, escalabilidad y resiliencia de la solución propuesta.

Implementar mecanismos de monitoreo y auditoría continua sobre los procesos de intercambio electrónico de información, incorporando registros de eventos (logs), alertas de seguridad y herramientas de análisis que permitan detectar oportunamente actividades sospechosas o incidentes relacionados con el envío y recepción de mensajes. Esta mejora contribuirá a fortalecer la capacidad de supervisión, trazabilidad y respuesta ante posibles amenazas de seguridad.

Evaluar la adopción de algoritmos criptográficos resistentes a la computación cuántica (Post-Quantum Cryptography), considerando las recomendaciones de organismos internacionales especializados en seguridad de la información. Esta línea de investigación permitirá preparar la infraestructura tecnológica de la organización frente a futuros escenarios donde los algoritmos criptográficos tradicionales puedan verse comprometidos por el avance de las tecnologías de computación cuántica.

7. BIBLIOGRAFIA

- Coloma Macias, R. L. (2021). Estudio comparativo de software basados en el cifrado y protección de datos (Bachelor's thesis, BABAHOYO: UTB, 2021
- Martinez, Miguel. (2020). Aplicación móvil Android para proteger información por medio de criptografía simétrica. 10.13140/RG.2.2.27833.39524.
- Riabov, V. V. (2005). SMTP (Simple Mail Transfer Protocol). River College.
- CHAVEZDODERO, Pgr CARLOS. ¿QUÉ ES EL CORREO ELECTRÓNICO? Quipukamayoc, 2014, vol. 7, no 14, p. 131-136.
- Larragan, M. G. (2016). Criptografía (XXXV): el algoritmo RSA (I). http://mikelgarcialarragan.blogspot.com/2016/10/criptografia-xxxv-el-algoritmo-rsa-i.html
- Winter, B., & Mollard, M. F. V. (2003). Gnu Privacy Guard (GnuPG) Mini Howto. retrieved from the Internet: http://www.gnupg.org/documentation/howtos.en.html, version 0.1, 4.
- Paredes, G. G. (2006). Introducción a la criptografía.
- Ramirez, J. C., Huerta, H. V., Rodriguez, C. R., & Bailón, F. E. (2020). Firma digital basada en criptografía asimétrica para generación de historial clínico. *3c Tecnología: glosas de innovación aplicadas a la pyme*, *9*(4), 65-85.
- Lino, J. C. B. LA CRIPTOGRAFÍA COMO PILAR DE LA SEGURIDAD INFORMÁTICA.
- Wehbe, R., Armas, A. D., & Barrera, E. (2022). Criptografía liviana para objetos conectados. In *XXIV Workshop de Investigadores en Ciencias de la Computación (WICC 2022, Mendoza)*.

- Martínez Pérez, S., Gómez Ramírez, J. P., López Vargas, C., & Moreno Álvarez, A. F. (2023). Fortalecimiento de la seguridad en aplicaciones web mediante criptografía avanzada: métodos y técnicas. *Revista Vínculos*, *20*(2).
- Lino, J. C. B. LA CRIPTOGRAFÍA COMO PILAR DE LA SEGURIDAD INFORMÁTICA.
- JIMENEZ NINA, J. O. S. U. E. (2022). *IMPLEMENTACIÓN DE ALGORITMO RSA PARA INCLUIR CONFIDENCIALIDAD EN APLICACIONES WEB CON EL USO DE HERRAMIENTA CRIPTOGRÁFICA* (Doctoral dissertation).
- Zenteno, A. C., Rubín, G. T., Pérez, J., & Márquez, E. Metodología para Cifrar Información en Aplicación de Mensajería. *María del Carmen Santiago Díaz*, 59.
- Lucas, G. I. C., Espinoza, R. E. G., De La Cruz, R. S. D., & Merino, M. J. M. (2022). Aplicación de protocolos SSL y TSL para él envió de información. *Journal TechInnovation*, *1*(2), 4-9.