

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE TELECOMUNICACIONES

TRABAJO DE INTEGRACIÓN CURRICULAR

TEMA:

"IMPACTO DE LA EXTENSIÓN DE SEGURIDAD DNSSEC SOBRE EL RENDIMIENTO DE SERVIDORES DNS FRENTE AL SISTEMA TRADICIONAL EN ENTORNOS SIMULADOS"

Trabajo de titulación previo a la obtención del título de Ingeniero en Telecomunicaciones

Línea de investigación: Desarrollo, aplicación de software y cyber security (seguridad cibernética)

AUTOR:

Brian Steve Rea Arias

DIRECTOR:

Ing. Hernán Mauricio Domínguez Limaico, Msc

Ibarra – Ecuador, 2026

UNIVERSIDAD TÉCNICA DEL NORTE
DIRECCIÓN DE BIBLIOTECA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO	
APELLIDOS Y NOMBRES:	REA ARIAS BRIAN STEVE

DATOS DE LA OBRA	
TÍTULO:	"IMPACTO DE LA EXTENSIÓN DE SEGURIDAD DNSSEC SOBRE EL RENDIMIENTO DE SERVIDORES DNS FRENTE AL SISTEMA TRADICIONAL EN ENTORNOS SIMULADOS"
AUTOR (ES):	BRIAN STEVE REA ARIAS
FECHA: DD/MM/AAAA	03/09/2026
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	■ **PREGRADO** ☐ **POSGRADO**
TITULO POR EL QUE OPTA:	INGENIERO EN TELECOMUNICACIONES
DIRECTOR /ASESOR:	ING. HERNÁN MAURICIO DOMÍNGUEZ LIMAICO, MSC ING. FABIÁN GEOVANNY CUZME RODRIGUEZ, MSC

2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 3 días del mes de septiembre de 2026

EL AUTOR:

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
BRIAN STEVE REA ARIAS

Brian Steve Rea Arias

CERTIFICACIÓN DEL DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Ibarra, 3 de septiembre de 2026

ING. HERNÁN MAURICIO DOMÍNGUEZ LIMAICO, MSC
DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

CERTIFICA:

Haber revisado el presente informe final del trabajo de Integración Curricular, el mismo que se ajusta a las normas vigentes de la Universidad Técnica del Norte; en consecuencia, autorizo su presentación para los fines legales pertinentes.



……

Ing. Domínguez Limaico Hernán Mauricio, Msc
C.C.: 1002379301

DEDICATORIA

Este trabajo de investigación está dedicado en primer lugar, a mi familia, por su apoyo incondicional, comprensión y motivación constante durante todo el proceso de formación académica, en especial a mis padres Jaime y Marcela que con su amor ha sido la fuerza más poderosa para no rendirme.

A mis docentes, quienes compartieron sus conocimientos, experiencias y orientación, contribuyendo al desarrollo de las capacidades necesarias para culminar este proyecto. Hago mención especial a mi director de tesis, Mauricio Domínguez por su inagotable paciencia conmigo y a mi asesor, Fabián Cuzme, por el tiempo y las valiosas lecciones que me brindó.

Al Ingeniero Jaime Michilena que, con su acertada orientación y la firme convicción de que podemos alcanzar la excelencia, logró impactar de manera profunda y positiva en mi vida académica.

Finalmente, dedico este trabajo a todas las personas interesadas en la seguridad informática y las redes de comunicación, con la esperanza de que esta investigación aporte al fortalecimiento del conocimiento sobre la implementación de tecnologías que mejoren la seguridad del Sistema de Nombres de Dominio (DNS).

Brian Steve Rea Arias

AGRADECIMIENTO

Expreso mi más sincero agradecimiento a Dios por brindarme salud, fortaleza y perseverancia para culminar esta etapa académica.

A mi familia, en especial a mis padres y mi tía Gabriela por su apoyo incondicional, su inmensa paciencia y la confianza depositada en mí durante el desarrollo de esta investigación.

A mis docentes y mi tutor de investigación, por sus acertadas observaciones y sus valiosos aportes, los cuales fueron fundamentales para fortalecer y enriquecer este trabajo.

Asimismo, agradezco a la Universidad Técnica del Norte por proporcionarme los recursos, el conocimiento y el ambiente académico necesarios para la realización de este estudio.

Finalmente, agradezco a todas las personas amigos, familia, compañeros que de una u otra manera, contribuyeron al desarrollo y culminación de esta investigación.

Brian Steve Rea Arias

RESUMEN

El Sistema de Nombres de Dominio (DNS) tradicional carece de mecanismos de autenticación del origen de los datos, lo que lo hace vulnerable a ataques de suplantación mediante respuestas forjadas. El presente trabajo evalúa de forma cuantitativa el compromiso entre seguridad y rendimiento que introduce la extensión DNSSEC frente al DNS tradicional. Para ello se implementó un entorno de simulación en GNS3 con routers Cisco C7200 emulados mediante Dynamips y servidores DNS desplegados en contenedores Docker con BIND 9.20 sobre Debian 12, organizados en tres zonas de red (LAN, tránsito y DMZ). Se definieron cuatro escenarios de evaluación: operación normal sin DNSSEC (E1) y con DNSSEC (E2), y operación bajo carga adversaria sin DNSSEC (E3) y con DNSSEC (E4), empleando un modelo de ataque determinista basado en la redirección controlada del reenvío del resolutor hacia un servidor autoritativo falso. Se midieron latencia, throughput, consumo de CPU y memoria, mediante las herramientas dig y dnsperf sobre un conjunto de diez mil nombres únicos. Los resultados muestran que, en operación normal, DNSSEC no introduce una penalización apreciable en la latencia (diferencias inferiores al 3 %), en el throughput (saturación en torno a 780 consultas por segundo, limitada por la red emulada) ni en el consumo de CPU y memoria del resolutor. El único costo diferencial y medible de DNSSEC se localizó en el volumen de datos por respuesta, que se incrementó de 57 a 177 bytes (factor aproximado de 3,1) por la inclusión de los registros de firma RRSIG, elevando la tasa de transferencia de 74,67 a 174,10 KB/s . En el plano de la seguridad, se verificó empíricamente que el resolutor con DNSSEC rechazó la respuesta forjada mediante el código SERVFAIL, mientras que el resolutor sin DNSSEC aceptó y almacenó la dirección falsificada.

Palabras clave: DNSSEC, seguridad DNS, suplantación de DNS, rendimiento de red, BIND, GNS3.

ABSTRACT

The traditional Domain Name System (DNS) lacks data-origin authentication mechanisms, making it vulnerable to spoofing attacks through forged responses. This work quantitatively evaluates the security-performance trade-off introduced by the DNSSEC extension compared to traditional DNS. A simulation environment was implemented in GNS3, using Cisco C7200 routers emulated with Dynamips and DNS servers deployed in Docker containers running BIND 9.20 on Debian 12, organized into three network zones (LAN, transit, and DMZ). Four evaluation scenarios were defined: normal operation without DNSSEC (E1) and with DNSSEC (E2), and operation under adversarial load without DNSSEC (E3) and with DNSSEC (E4), using a deterministic attack model based on the controlled redirection of the resolver's forwarding toward a rogue authoritative server. Latency, throughput, CPU and memory consumption, and data volume were measured using the dig and dnsperf tools over a set of ten thousand unique names. The results show that, under normal operation, DNSSEC introduces no appreciable penalty in latency (differences below 3 %), throughput (identical saturation at approximately 780 queries per second, limited by the emulated network), or the resolver's CPU and memory consumption. The only measurable differential cost of DNSSEC was found in the data volume per response, which increased from 57 to 177 bytes (a factor of approximately 3.1) due to the inclusion of RRSIG signature records, raising the transfer rate from 74.67 to 174.10 KB/s (a factor of 2.33). Regarding security, it was empirically verified that the DNSSEC-enabled resolver rejected the forged response with a SERVFAIL code, preventing cache poisoning, whereas the resolver without DNSSEC accepted and cached the falsified address.

Keywords: DNSSEC, DNS security, DNS spoofing, network performance, BIND, GNS3.

LISTA DE CONTENIDOS

1. Capítulo I: Antecedentes 17

1.1 Tema 21

1.2 Problema 21

1.3 Objetivos 25

1.3.1. Objetivo General 25

1.3.2. Objetivos Específicos 25

1.4 Alcance 25

1.5 Justificación 29

2. Capítulo II: Estado del Arte 31

2.1 Sistema de nombres de Dominio (DNS) 31

2.1.1. Estructura Jerárquica del DNS 34

2.1.2. Funcionamiento del DNS Tradicional 36

2.1.3. Vulnerabilidades del DNS 39

2.2 Extensión de Seguridad DNSSEC. 40

2.2.1. Componentes de la extensión DNSSEC 41

2.2.2. Cadena de Confianza y Validación. 43

2.2.3. RFC 6841: Declaraciones de Políticas y Prácticas. 46

2.3 Privacidad en DNS: Desafíos y Soluciones. 47

2.3.1. Riesgo de Exposición de Datos 47

2.3.2. Protocolos de Cifrado (DoH/DoT)..... 48

2.3.3. Marco Regulatorio. 49

3. Capítulo III: Diseño e implementación..... 51

3.1 Enfoque y Metodología de Investigación 51

3.2 Requerimientos. 55

3.2.1. Requerimientos de Stakeholders (StSR)..... 55

3.2.2. Requerimientos del Sistema (SySR)..... 56

3.2.3. Requerimientos de Software (SRS) 59

3.2.4. Especificación de Requerimientos del Modelo de Análisis (MRS)..... 60

3.2.5. Especificación de Requerimientos de Arquitectura (SyRA)..... 61

3.3 Arquitectura del Entorno Experimental..... 63

3.3.1. Selección del Software de Simulación..... 63

3.3.2. Diseño de la Topología de Red. 65

3.4 Implementación del Entorno de simulación 69

3.4.1. Especificaciones del Entorno. 69

3.4.2. Segmentación de red..... 71

3.4.3. Justificación de la Topología de Red 78

3.5 Implementación de los Nodos..... 80

3.5.1. Configuración del enrutamiento 80

3.5.2. Configuración del servidor DNS autoritativo. 84

3.5.3. Configuración DNS esclavo. 92

3.5.4. Configuración del servidor DNS recursivo..... 93

3.5.5. Servidor Web Corporativo. 96

3.5.6. Nodo Analizador de Tráfico. 97

3.5.7. Implementación de DNSSEC 98

3.5.8. Módulo de ataque implementado. 103

3.5.9. Herramientas de medición 106

3.6 Ejecución de pruebas 107

3.6.1. Escenarios experimentales 107

3.6.2. Métricas evaluadas 108

3.6.3. Procedimiento de Medición de Escenarios 109

3.7 Resultados experimentales 110

3.7.1. Escenario 1 - DNS tradicional sin ataque (línea base) 110

3.7.2. Configuración del Escenario 112

3.7.3. Escenario 2 - DNS con DNSSEC sin ataque 117

3.7.4. Configuración del Escenario 2 119

3.7.5. Procedimiento de Medición del Escenario 2 120

3.7.6. Escenario 3 - DNS tradicional bajo carga Adversaria..... 123

3.7.7. Escenario E3a – Rendimiento sin DNSSEC bajo carga adversaria. 125

3.7.8. Escenario E3b-Aceptación de la respuesta forjada sin DNSSEC. 126

3.7.9. Escenario 4 - DNS con DNSSEC bajo carga adversaria 129

3.7.10. Escenario 4a - Rendimiento bajo carga adversaria con DNSSEC 130

3.7.11. Escenario 4b- Rechazo de la respuesta forjada. 131

3.7.12. Throughput bajo carga progresiva 133

3.7.13. Uso de recursos del sistema (CPU/RAM) 140

4. CAPÍTULO IV: ANALISIS Y EVALUACIÓN DE RESULTADOS 142

4.1 Rendimiento en condiciones normales (Escenarios 1 y 2) 142

4.2 Comportamiento bajo carga adversaria (Escenarios 3 y 4)..... 145

4.2.1. Escenario E3a - rendimiento sin DNSSEC bajo carga adversaria. 145

4.2.2. Escenario E3b - Aceptación de respuesta forjada en DNSSEC. 146

4.2.3. Escenario E4a - Rendimiento con DNSSEC bajo carga adversaria 147

4.2.4. Escenario E4b – Rechazo de la respuesta forjada con DNSSEC..... 149

4.3 Análisis comparativo integral 151

4.4 Análisis del Punto de Saturación del Throughput. 153

4.4.1. Interpretación del hallazgo. 155

4.4.2. Throughput bajo carga adversaria (E3a/E4a)..... 155

4.4.3. Consumo de recursos del sistema (CPU / memoria) 156

4.5 Validación y cumplimiento de estándares 158

4.5.1. Alineación con NIST SP 800-81r3 158

4.6 Incremento en el tamaño de paquetes y riesgo de fragmentación 159

4.7 Cumplimiento del RFC 6841 161

4.7.1. Síntesis del compromiso seguridad - rendimiento 161

4.7.2. Limitaciones del estudio 162

CONCLUSIONES 163

RECOMENDACIONES 166

5. Bibliografía 168

ÍNDICE DE ANEXOS 171

Anexo A. Topología y direccionamiento de la red 171

A.1 Diagrama de la topología 172

Anexo B. Configuración de los servidores DNS 172

B.1 Servidor DNS Autoritativo (192.168.30.20) 172

B.1.1 Archivo de zona empresa.local 174

B.1.2 Claves DNSSEC 175

B.1.3 Estadísticas de la zona firmada 175

B.2 Servidor DNS Recursivo (192.168.20.10) 175

Anexo C. Scripts del nodo Atacante (192.168.10.66) 177

C.1 dns_falso.py - Servidor autoritativo malicioso 177

C.2 generar_ataque.sh - Generador de carga adversaria 178

Anexo D. Scripts de medición. 180

D.1 medir.sh - Latencia, throughput y recursos 180

D.2 verificar-cache.sh - Monitor del estado de la caché 181

Anexo E. Cumplimiento del NIST SP 800-81r3 182

Anexo F. RFC’s aplicados en la implementación. 183

LISTA DE FIGURAS

Figura 1 *Árbol de problemas sobre el problema de baja adopción de DNSSEC en Ecuador.* 24

Figura 2 *Escenario de envenenamiento de caché en DNS tradicional.* 28

Figura 3 *Delegación de Zonas dentro de una Empresa Pública.* 33

Figura 4 *Jerarquía de Dominios DNS* 35

Figura 5 *Pasos de resolución DNS.* 37

Figura 6 *Cadena de Confianza y Validación de DNSSEC* 44

Figura 7 *Fases de la Metodología implementada* 54

Figura 8 *Arquitectura de red implementada en el simulador GNS3* 67

Figura 9 *Topología de red implementada en GNS3 con segmentación por zonas.* 70

Figura 10 *Zona Transit donde se aloja el Servidor Recursivo y el Analizador de la simulación.* 72

Figura 11 *Zona Backbone* 73

Figura 12 *Zona LAN Donde se representa los usuarios corporativos, el tráfico legítimo y la amenaza interna.* 74

Figura 13 *Componentes que se encuentran en la Zona DMZ (servidores).* 75

Figura 14 *Tabla de enrutamiento del router R1-BORDE* 81

Figura 15 *Tabla de enrutamiento del router R1-CORE* 82

Figura 16 *Estado de las interfaces del router R1-BORDE.* 82

Figura 17 *Estado de las interfaces del router ISP-CORE* 83

Figura 18 *Validación de la zona empresa.local con named-checkzone.* 89

Figura 19 *Transferencia de Zona recibida en el Servidor Esclavo.* 89

Figura 20 *Verificación de resolución DNS correcta desde el cliente.* 90

Figura 21 *Zona empresa.local firmada.* 90

Figura 22 *Zona Firma con RRSIG.* 91

Figura 23 *Conteo de registros DNSSEC.* 92

Figura 24 *Página web de la empresa.local en html.* 97

Figura 25 *Firma exitosa de la zona empresa.local con DNSSEC* 100

Figura 26 *Zona empresa.local firmada.* 100

Figura 27 *Zona firmada con RRSIG.* 101

Figura 28 *Conteo de registros DNSSEC.* 101

Figura 29 *Registro DS de la zona empresa.local.* 102

Figura 30 *Verificación de la validación DNSSEC (flag AD).* 103

Figura 31 *Escenario 1 comportamiento normal de la infraestructura frente a distintos estados de la caché del servidor recursivo* 111

Figura 32 *Resolución en E1: la respuesta incluye RRSIG pero el recursivo no valida (ausencia del flag AD)* 115

Figura 33 *Comparación de latencia y throughput entre caché frío y caliente en E1.* ...115

Figura 34 *Análisis de PCAP en el Servidor Recursivo.* 116

Figura 35 *Escenario 2 Validación DNSSEC Habilitada y Ancla de Confianza configurada* 118

Figura 36 *Verificación de DNSSEC activo en E2* 120

Figura 37 *Comparación de latencia entre caché frío y caliente en E2.* 121

Figura 38 *Medición bajo tráfico concurrente con DNSSEC activo y rendimiento estable* 122

Figura 39 *Explotación de vulnerabilidad de protocolo mediante carga adversaria en DNS tradicional (Escenario 3).* 124

Figura 40 *Verificación del envenenamiento en E3b mediante consultas repetidas.* 127

Figura 41 *Aceptación de la respuesta forjada en E3b: detalle de la resolución.* 128

Figura 42 *Mitigación del ataque mediante validación criptográfica DNSSEC (Escenario 4).* 129

Figura 43 *Rechazo de la respuesta forjada en E4b.* 132

Figura 44 *Captura de paquetes de E4b mostrando la respuesta forjada del atacante y el rechazo del recursivo.* 133

Figura 45 *Comparación de latencia entre E1 y E2 (caché frío y caliente).* 144

Figura 46 *Aceptación de respuesta forjada en E3b* 147

Figura 47 *Rechazo de la respuesta forjada en E4b* 150

Figura 48 *Comparación de latencia y CPU entre los cuatro escenarios.* 153

Figura 49 *Punto de Saturación entre Cliente y Analizador.* 154

Figura 50 *Consumo de CPU del recursivo por escenario.* 158

Figura 51 *Comparación del tamaño de paquetes DNS con y sin DNSSEC.* 160

Figura 52 *Topología de red del entorno experimental en GNS3.* 172

LISTA DE TABLAS

Tabla 1 *Tipos de ataques y Mitigación mediante DNSSEC* 47

Tabla 2 *Requerimientos de Stakeholders* 56

Tabla 3 *Requerimientos del Sistema (SySR)* 56

Tabla 4 *Requerimientos de Software (SRS)* 59

Tabla 5 *Requerimientos del Modelo de Análisis* 60

Tabla 6 *Requerimientos de Arquitectura* 61

Tabla 7 *Herramientas de Evaluación de Redes.* 64

Tabla 8 *Dispositivos que conforman la topología de simulación.* 67

Tabla 9 *Información de Componentes y su especificación.* 70

Tabla 10 *Composición de cada zona con su segmento de red y la función que cumple.* . 76

Tabla 11 *Direccionamiento IP completo de la topología.* 77

Tabla 12 *Tabla de verificación de conectividad.* 83

Tabla 13 *Interpretación de los campos temporales del registro SOA.* 86

Tabla 14 *Dominios configurados en la zona empresa.local (90 registros).* 86

Tabla 15 *Directivas clave del servidor recursivo frente al autoritativo.* 95

Tabla 16 *Escenarios experimentales definidos.* 107

Tabla 17 *Resultados del Escenario 1 - DNS tradicional sin ataque.* 113

Tabla 18 *Uso de recursos del sistema con DNSSEC activo.* 123

Tabla 19 *Resultados del Escenario 3a DNS sin DNSSEC bajo carga adversaria.* 125

Tabla 20 *Contraste de validación DNSSEC ante la respuesta forjada (E3b sin DNSSEC / E4b con DNSSEC).* 127

Tabla 22 *Comparación de la validación con y sin DNSSEC ante la respuesta forjada.* 131

Tabla 23 *Throughput en bytes desde el cliente – E1 (sin DNSSEC).* 135

Tabla 24 *Throughput en bytes desde el cliente – E2 (con DNSSEC, bit DO).* 135

Tabla 25 *Throughput en bytes desde el analizador – E1 (sin DNSSEC).* 136

Tabla 26 *Throughput en bytes desde el analizador – E2 (con DNSSEC, bit DO).* 136

Tabla 27 *Throughput legítimo en bytes bajo ataque – E3a (sin DNSSEC).* 137

Tabla 28 *Throughput legítimo en bytes bajo ataque – E4a (con DNSSEC).* 138

Tabla 29 *Comparación del punto de saturación según el punto de medición (DNSSEC activo).* 139

Tabla 30 *Uso de recursos del sistema por contenedor.* 140

Tabla 31 *Resultados del Escenario 1 - DNS tradicional sin ataque.* 143

Tabla 32 *Resultados del Escenario 3 - ataque sin DNSSEC.* 145

Tabla 33 *Resultados del Escenario 4a - DNS con DNSSEC bajo carga adversaria.* 148

Tabla 34 *Comparación de la validación con y sin DNSSEC ante la respuesta forjada.* 149

Tabla 35 *Consolidación de métricas por escenario.* 152

Tabla 36 *Uso de recursos por escenario.* 157

Tabla 37 *Validación contra el NIST SP 800-81r3.* 159

Tabla 38 *Síntesis del compromiso seguridad - rendimiento.* 161

Tabla 39 *Índice de anexos del trabajo de titulación.* 171

Tabla 40 *Tabla de cumplimiento del NIST SP 800-81r3* 182

Tabla 41 *RFC's en la implementación.* 183

LISTA DE BLOQUES DE CÓDIGO

Bloque de Código 1 *Declaración de la zona empresa.local en named.conf.local.* 84

Bloque de Código 2 *Opciones relevantes en named.conf.options.* 84

Bloque de Código 3 *Extracto del archivo de zona db.empresa.local.* 85

Bloque de Código 4 *Comandos de validación de la zona y la configuración.* 88

Bloque de Código 5 *Verificación de la transferencia de zona hacia el esclavo.* 89

Bloque de código 6 *Configuración del origen de la transferencia en el servidor Esclavo.* 92

Bloque de código 7 *Configuración del recursivo named.conf.options.* 94

Bloque de código 8 *Zona de reenvío hacia el servidor autoritativo.* 94

Bloque de código 9 *Ancla de confianza estática para la zona empresa.local.* 95

Bloque de código 10 *Comandos de gestión del caché del resolutor.* 96

Bloque de Código 11 *Validación de la configuración y comprobación de resolución.* ... 96

Bloque de código 12 *Generación de claves y firma de la zona.* 98

Bloque de código 13 *Trust anchor en el servidor recursivo.* 99

Bloque de código 14 *Fragmento de dns_falso.py.* 105

Bloque de código 15 *Configuración de named.conf* 173

Bloque de código 16 *Configuración de named.conf.options en el Servidor Autoritativo.* 173

Bloque de código 17 *Configuración del codigo named.conf.local* 173

Bloque de código 18 *Configuración de zonas db.empresa.local* 174

Bloque de código 19 *Claves de KSK y ZSK* 175

Bloque de código 20 *Estadísticas de la zona firmada.* 175

Bloque de código 21 *Configuración del archivo named.conf del Servidor Recursivo* .. 176

Bloque de código 22 *Configuración del Archivo named.conf.options del Servidor Recursivo.* 176

Bloque de código 23 *Configuración del archivo named.conf.local del Servidor Recursivo.* 176

Bloque de código 24 *Archivo con el script del dns_falso.py* 177

Bloque de código 25 *Archivo generar_ataque* 178

Bloque de código 26 *Scripts para medir latencia, throughput y recursos.* 180

Bloque de código 27 *Archivo .sh del Monitor del estado de caché.* 181

1. Capítulo I: Antecedentes

1.1 Tema

Impacto de la extensión de seguridad DNSSEC sobre el rendimiento de servidores DNS frente al sistema tradicional en entornos simulados.

1.2 **Problema**

En el entorno global, la ciberseguridad enfrenta retos crecientes debido a la evolución de técnicas de ciberataques y la dependencia critica de infraestructuras digitales. Según Huston Geoff, (2023), en 2022 se registraron más de 1.2 millones de intentos de ciberataques a nivel mundial, con un aumento del 38% en ataques dirigidos a gobiernos y servicios esenciales. Entre las vulnerabilidades estructurales más preocupantes se encuentra el Sistema de Nombres de Dominio (DNS), diseñado en la década de 1980 sin mecanismos nativos de autenticación robusta. Esto ha facilitado ataques como el envenenamiento de caché DNS, donde actores maliciosos redirigen tráfico a servidores fraudulentos, comprometiendo la integridad de la información según (Southam, (2014).

La implementación de DNSSEC (Extensiones de Seguridad para DNS), desarrollada para mitigar estas vulnerabilidades mediante firmas digitales y autenticación jerárquica, ha sido desigual a nivel global. Mientras países como Suecia y Países Bajos superan el 90% de adopción de DNSSEC en sus dominios nacionales, regiones como América Latina y África presentan tasas inferiores al 25% (El Comercio, 2019). Esta disparidad expone a millones de usuarios a riesgos evitables, especialmente en economías en desarrollo donde la concienciación y los recursos técnicos son limitados (Akanho y Muchene, 2021).

A nivel regional, países como Colombia han iniciado programas para fomentar la implementación de DNSSEC entre los ISPs, logrando que aproximadamente el 21.74% de las solicitudes DNS sean autenticadas con DNSSEC. No obstante, este porcentaje indica que la mayoría del tráfico DNS aún no cuenta con esta capa adicional de seguridad (NAP Colombia, (2024).

Ecuador enfrenta un panorama crítico en materia de ciberseguridad así es como en 2023, el país registró más de 12 millones de ciberataques**,** según un estudio de Movistar Empresas, lo que refleja una disminución del 27% respecto a 2022. Sin embargo, esta cifra sigue siendo alarmante especialmente porque los atacantes han adoptado estrategias más sofisticadas, como extorsiones dirigidas a empresas y entidades públicas.(Teleamazonas, 2023).

Este contexto se alinea con lo expuesto por Juca-Maldonado y Medina-Peña, (2023) en su estudio de Ciberdelitos en Ecuador y su impacto social, en el que destacan que el robo de información personal, fraude en línea y ataques a infraestructuras críticas son las amenazas más recurrentes, con un aumento del 124% en incidentes durante la pandemia. Un ejemplo emblemático que muestra la vulnerabilidad de los sistemas, es el ataque a la Contraloría General del Estado en 2019 que comprometió más de 16.000 cuentas de correo (El Comercio, 2019).

Este panorama resalta la necesidad urgente de reforzar los mecanismos de seguridad del DNS. El Sistema de Nombres de Dominio (DNS) es fundamental para el funcionamiento de Internet, pero su diseño original carece de mecanismos de seguridad robustos, lo que hace ser susceptible a los ataques como el envenenamiento de caché. Este tipo de ataque permite a los agresores redirigir el tráfico hacia sitios maliciosos, comprometiendo la integridad y confidencialidad de la información según Armas Gutiérrez, (2017).

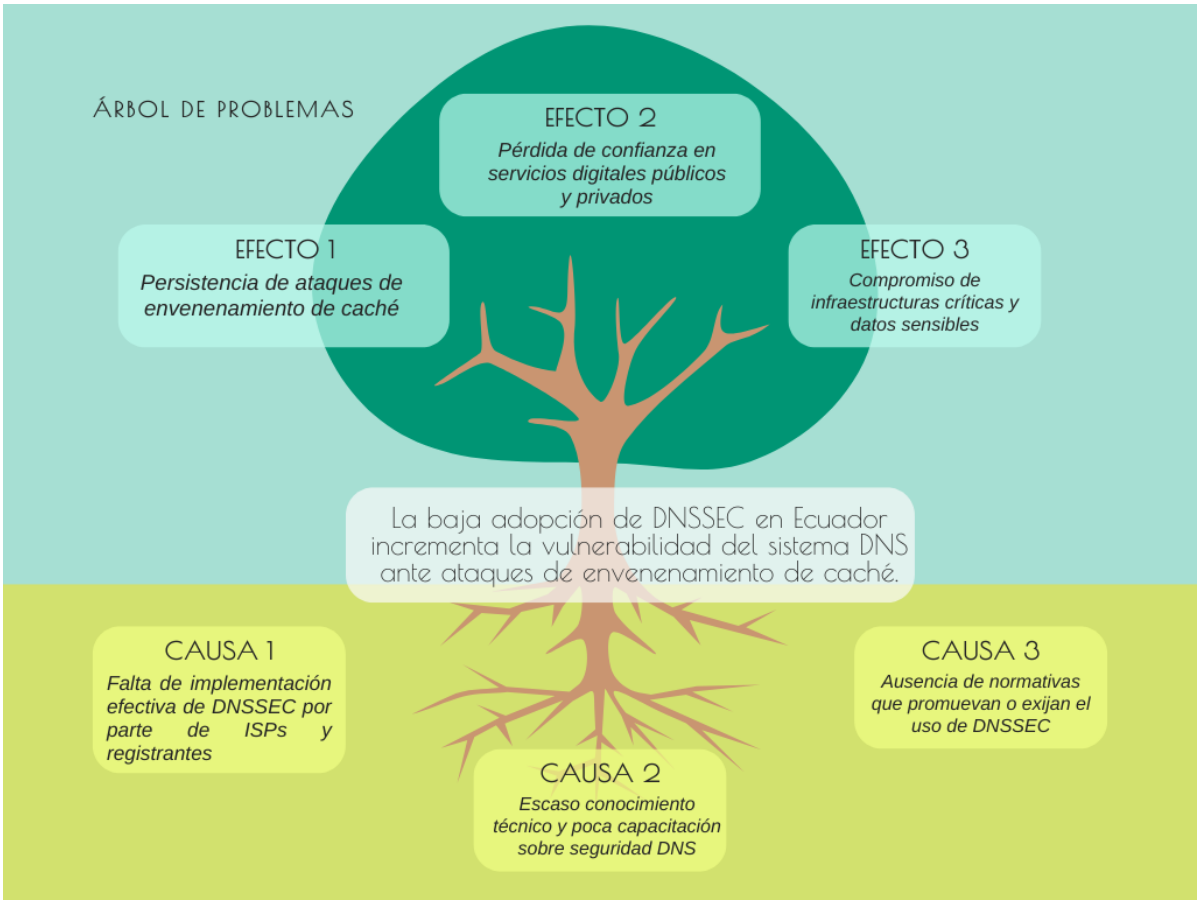
Para mitigar estas vulnerabilidades, se desarrolló DNSSEC (Extensiones de Seguridad para DNS), que añade autenticación y verificación de integridad a las respuestas DNS mediante firmas digitales. Sin embargo, la adopción de DNSSEC en América Latina ha sido limitada y desigual.

En Ecuador, por ejemplo, aunque el dominio de nivel superior .ec fue firmado con DNSSEC en el segundo semestre de 2023, la implementación efectiva por parte de los registrantes y proveedores de servicios de Internet (ISPs) aún es incipiente; esta situación deja a una parte significativa de los usuarios de Internet en el país expuestos a ataques que podrían ser prevenidos con la adopción de DNSSEC.

La falta de adopción generalizada de DNSSEC en América Latina y específicamente en Ecuador, plantea una preocupación significativa en términos de ciberseguridad, así como también la ausencia de mecanismos de autenticación en el DNS facilita la ejecución de ataques de envenenamiento de caché lo que pone en riesgo la confianza y seguridad de los usuarios en la infraestructura de Internet.

Figura 1

Árbol de problemas sobre el problema de baja adopción de DNSSEC en Ecuador.



1.3 Objetivos

1.3.1. *Objetivo General*

Evaluar el impacto de DNSSEC en el rendimiento de servidores DNS mediante un análisis comparativo de métricas cuantitativas frente al DNS tradicional en entornos simulados controlados.

1.3.2. *Objetivos Específicos*

- Estudiar las estructuras y las características fundamentales del DNS tradicional como la arquitectura jerárquica, vulnerabilidades de caché y DNSSEC tales como firmas RRSIG, claves DNSKEY, y cadena DS; analizando sus diferencias en seguridad y funcionamiento.
- Implementar un entorno de simulación con servidores DNS tradicional y DNSSEC que permita emular ataques de envenenamiento de caché para generar cargas de tráfico controladas para medir métricas cuantitativas.
- Comparar cuantitativamente el rendimiento de DNS tradicional y DNSSEC mediante métricas como latencia, throughput y uso de recursos del sistema, tanto en condiciones normales como bajo escenarios de ataque y carga variable.
- Interpretar los resultados obtenidos para determinar el impacto de DNSSEC en entornos simulados identificando fortalezas, debilidades y compromisos entre seguridad y rendimiento con base en los datos experimentales recolectados.

1.4 Alcance

El presente trabajo se enmarca en el análisis del impacto de la extensión de seguridad del sistema de nombres de dominio (DNSSEC) en el rendimiento de servidores DNS, comparándola con implementaciones tradicionales sin mecanismos de seguridad sino a través de un entorno de simulación controlado se estudiarán métricas clave como latencia, tasa de transferencia y uso de

recursos (CPU, memoria, red) para determinar las implicaciones de integrar DNSSEC en sistemas de resolución de nombres.

La investigación busca cuantificar cómo las operaciones criptográficas y los mecanismos adicionales de validación introducidos por DNSSEC afectan el desempeño operativo de los servidores DNS en comparación con DNS tradicional. El estudio permitirá observar la diferencia de rendimiento tanto en servidores autoritativos como en resolutores recursivos, bajo distintos niveles de carga.

El entorno de validación estará basado en una arquitectura virtualizada, en la cual se emularán servidores DNS primarios y resolutores recursivos, ejecutando escenarios con y sin DNSSEC habilitado. Los ataques de envenenamiento de caché serán simulados para observar no solo la respuesta frente a amenazas, sino también la carga adicional generada en el procesamiento de peticiones DNS seguras.

La metodología empleada en esta investigación es de tipo Investigación Experimental Validada, estructurada en seis fases secuenciales, cada una de las cuales contribuye al cumplimiento progresivo de los objetivos específicos planteados.

La primera fase corresponde a la Planificación que contempla el diseño del entorno de simulación y la definición clara de los objetivos del estudio, alineándose con el primer objetivo específico.

En la segunda fase Construcción, en esta se procede a la configuración del sistema DNS tradicional y DNSSEC dentro de un entorno virtualizado, con parámetros estandarizados para garantizar comparabilidad, lo que cumple con el segundo objetivo específico.

La tercera fase, Ejecución que incluye la simulación de ataques como el envenenamiento de caché, así como la generación de tráfico y la recolección de métricas de rendimiento, contribuyendo al tercer objetivo específico.

La cuarta fase de Análisis, se procesan los datos obtenidos mediante herramientas estadísticas para evaluar las diferencias de comportamiento entre ambos sistemas, lo que se relaciona con el cuarto objetivo.

La quinta fase Validación, implica la revisión y comparación de los resultados obtenidos con marcos normativos internacionales, asegurando la coherencia con buenas prácticas establecidas.

Finalmente, la sexta fase de Documentación en la que se elabora el informe final y se formulan recomendaciones prácticas basadas en los hallazgos de la investigación.

La presente investigación se apoya en marcos normativos reconocidos que permiten establecer una base técnica y metodológica para el análisis del impacto de DNSSEC en el rendimiento de servidores DNS. En particular, se consideran dos referencias clave: el RFC 6841 Ljunggren et al., (2013)y la guía NIST SP 800-81r2 Chandramouli & Rose, (2013) .Para analizar el impacto de la implementación de DNSSEC en comparación con el sistema DNS tradicional, se plantea el diseño y despliegue de una red virtual controlada utilizando plataformas de simulación, tales como GNS3, ns-3 o entornos basados en máquinas virtuales con Linux. Esta infraestructura permitirá reproducir condiciones reales de operación, manteniendo el control de variables técnicas para asegurar la validez de los resultados.

En este entorno simulado, se configurarán servidores DNS tanto con configuraciones convencionales como con DNSSEC habilitado. La propuesta prioriza el uso de mecanismos de seguridad eficientes, con el objetivo de evaluar su desempeño sin introducir cargas operativas

innecesarias. Las configuraciones serán estandarizadas para permitir una comparación equitativa entre ambos enfoques.

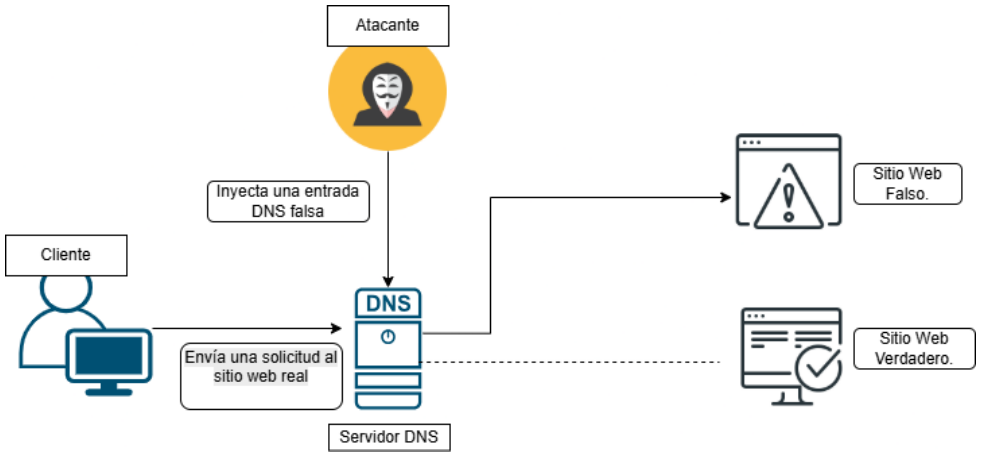
A través de pruebas de laboratorio reproducibles, se medirán métricas cuantitativas clave como: Tiempo promedio de respuesta a consultas DNS, Capacidad de manejo de solicitudes simultáneas (concurrencia), Consumo de recursos del sistema (CPU, memoria y red).

Estas métricas permitirán evaluar el rendimiento operativo de los servidores en ambas modalidades. Además, se incluirán escenarios de riesgo controlados, como ataques de envenenamiento de caché o manipulación de respuestas DNS; con el fin de examinar la capacidad de mitigación y detección que ofrece DNSSEC frente a DNS convencional.

Los resultados obtenidos servirán como base para identificar fortalezas, limitaciones y compromisos de rendimiento asociados al uso de DNSSEC, proporcionando una visión integral de su aplicabilidad en entornos reales.(Guzzo, 2024).

Figura 2

Escenario de envenenamiento de caché en DNS tradicional.



Nota: Este escenario puede variar según las necesidades para identificar el valor cuantitativo de los valores de un servidor DNS.

La Figura 2 muestra un entorno básico de resolución de nombres DNS tradicional, el cual será replicado en el entorno experimental propuesto en esta investigación. En esta configuración, un cliente realiza una consulta legítima a un servidor DNS con el objetivo de acceder a un sitio web auténtico. No obstante, debido a la ausencia de mecanismos de autenticación en el protocolo DNS convencional, un atacante puede aprovechar esta vulnerabilidad para inyectar una entrada falsa en la caché del servidor resolutivo. Esto se logra mediante técnicas como spoofing o respuesta anticipada(Chandrasekaran et al., 2023). Como resultado, el servidor DNS responde con una dirección IP manipulada que redirige al cliente hacia un sitio web falso, comprometiendo así la integridad del proceso de resolución de nombres. Esta simulación forma parte del escenario controlado que se utilizará para evaluar el impacto de DNSSEC en la mitigación de este tipo de ataques.

1.5 Justificación

La presente investigación nos permite estudiar las estructuras y características del DNS tradicional y DNSSEC para fortalecer la ciberseguridad, se contribuye al ODS 9 que es de Industria, Innovación e Infraestructura (Naciones Unidas, 2015) mediante el análisis de infraestructuras digitales resilientes, y al ODS 16 Paz, Justicia e Instituciones Sólidas (Naciones Unidas, 2015) al proponer mecanismos que protejan sistemas críticos contra ciberataques, asegurando servicios públicos confiables.

La investigación propone una evaluación técnica integral de la implementación de DNSSEC, comparada con el DNS tradicional, considerando aspectos tanto de seguridad como de rendimiento operativo; analizando específicamente las vulnerabilidades relacionadas con la

cadena de confianza, como la incorrecta propagación de registros DS o errores en la rotación de claves KSK/ZSK, tomando como marco de referencia el RFC 6841 Kolkman, (2013) el cual proporciona un estándar para desarrollar políticas y prácticas de implementación DNSSEC (DPS). La identificación de fallas estructurales en este componente es clave para determinar el verdadero grado de robustez de una arquitectura DNS segura Chandrasekaran et al., (2023).

Se validarán las configuraciones de seguridad de los servidores DNS autoritativos y resolutivos implementados, contrastándolas con las prácticas recomendadas en el NIST SP 800-81r2 Chandramouli & Rose, (2013b). De esta manera detectar desviaciones en estas configuraciones permitirá generar recomendaciones concretas para fortalecer la infraestructura DNS en entornos reales.

El aporte de esta investigación radica en que, se desarrollará un modelo de implementación de DNSSEC adaptado a entornos simulados, que integre prácticas seguras y eficientes basadas en estándares internacionales. En segundo lugar, se generarán insumos técnicos validados empíricamente que podrán ser aprovechados por entidades públicas, financieras y educativas del Ecuador. Este estudio; por tanto, contribuirá al fortalecimiento de la ciberseguridad nacional, y también proporcionará un marco de referencia útil para futuras auditorías y despliegues de DNSSEC.

2. Capítulo II: Estado del Arte

En el segundo capítulo se realiza un análisis bibliográfico integral sobre el Sistema de Nombres de Dominio (DNS) y sus vulnerabilidades estructurales, se examina la arquitectura jerárquica del protocolo, su funcionamiento tradicional y las falencias de seguridad que lo afectan como es el envenenamiento de caché y ataques de amplificación. Posteriormente, se introduce la extensión DNSSEC como solución técnica, se desglosan sus componentes criptográficos (RRSIG, DNSKEY, DS), el mecanismo de cadena de confianza, y se contrastan sus beneficios frente a limitaciones operativas. Por otra parte, la investigación profundiza en un análisis teórico sobre DNSSEC y (DoH/DoT), de igual manera junto con los estándares normativos que rigen su implementación y buenas prácticas.

2.1 Sistema de nombres de Dominio (DNS).

El DNS (Domain Name System) es un sistema distribuido y jerárquico que, según Dooley y Rooney,(2017), opera como "la piedra angular de la conectividad en Internet", y funciona de manera análoga a una agenda telefónica global. Los autores destacan que su propósito esencial es "traducir nombres de dominio legibles por humanos como www.google.com en direcciones IP numéricas (como 142.250.190.46); permitiendo así la comunicación entre dispositivos"Dooley y Rooney, (2017). Esta funcionalidad, resulta fundamental para la experiencia de usuario, ya que "sin el DNS, los usuarios se verían obligados a memorizar secuencias numéricas complejas para acceder a cualquier sitio web o servicio en línea" (Dooley y Rooney, 2017, p. 19).

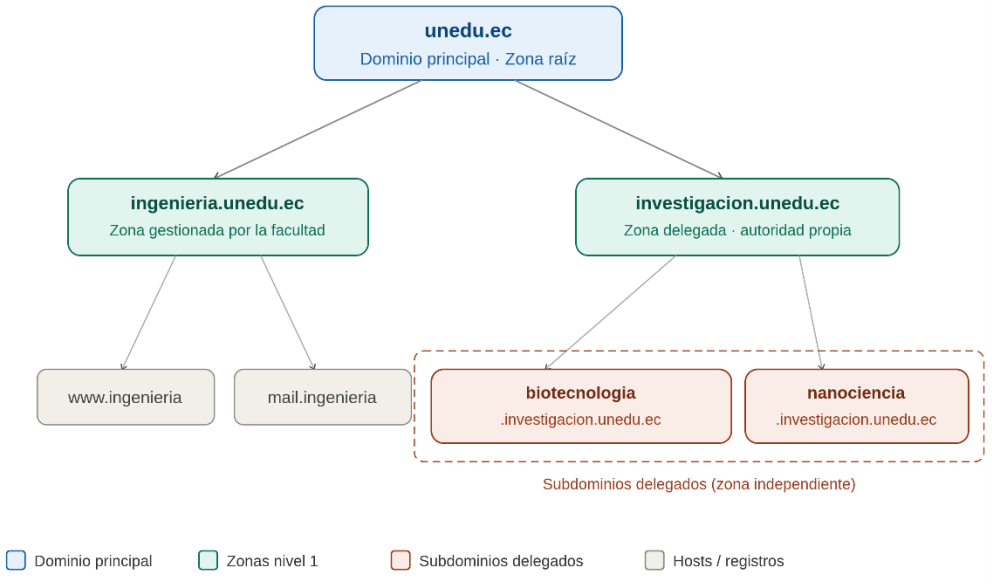
Es necesario recordar que el DNS opera como una base de datos distribuida globalmente organizada como una multinacional tecnológica, donde la autoridad se delega desde el liderazgo superior hasta los equipos operativos. Esta jerarquía consta de cuatro niveles fundamentales que garantizan la escalabilidad del sistema. En la cúspide se encuentra la zona raíz (.), administrada

por 13 conjuntos de servidores gestionados por organizaciones como ICANN y VeriSign. Estos servidores dirigen las consultas hacia los dominios de primer nivel (TLD), que se clasifican en genéricos (.com, .org) y territoriales (.co, .ec). Por debajo, en segundo lugar, los dominios de segundo nivel (Second Level Domain) representan nombres registrados por entidades específicas (ej: "google" en google.com), mientras que en tercer lugar los subdominios permiten segmentar servicios (mail.google.com). Cada nivel depende de servidores autoritativos que almacenan los registros DNS (A, MX, CNAME) y responden consultas para sus zonas delegadas. Esta descentralización evita cuellos de botella, pero introduce complejidad en la gestión coordinada de la infraestructura global (Hoffman et al., 2019).

Es importante conocer las zonas y dominios ya que estos forman parte de la estructura jerárquica la cual se explicará con más detalle, por lo que el término zona define el alcance del control administrativo delegado dentro de esa jerarquía. Por ejemplo, en una universidad ecuatoriana la zona principal unedu.ec tendría autoridad sobre los dominios de sus facultades (ingenieria.unedu.ec, medicina.unedu.ec).

Figura 3

Delegación de Zonas dentro de una Empresa Pública.



Como se observa en la Figura 3 una zona delegada como investigacion.unedu.ec gestionaría autónomamente su propio dominio y todos sus subdominios asociados (biotecnologia.investigacion.unedu.ec, nanociencia.investigacion.unedu.ec), demarcando así niveles específicos de gestión dentro del sistema global de nombres.

Para mantener la coherencia entre servidores autoritativos, se utilizan transferencias de zona (AXFR/IXFR)[1]. Por ejemplo, los servidores esclavos de unedu.ec sirven para sincronizar cambios. en zonas dinámicas actualizadas por DHCP[2] como en redes de campus universitarios, pero se prefiere transferencias incrementales (IXFR).

[1] Mecanismo de sincronización en DNS donde:
AXFR (Transferencia Completa): Transfiere *toda* la zona DNS.
IXFR (Transferencia Incremental): Solo envía cambios desde la última sincronización, que es ideal para zonas con actualizaciones frecuentes.

[2] **DHCP** es el protocolo que asigna direcciones IP dinámicamente a dispositivos, generando constantes actualizaciones DNS.

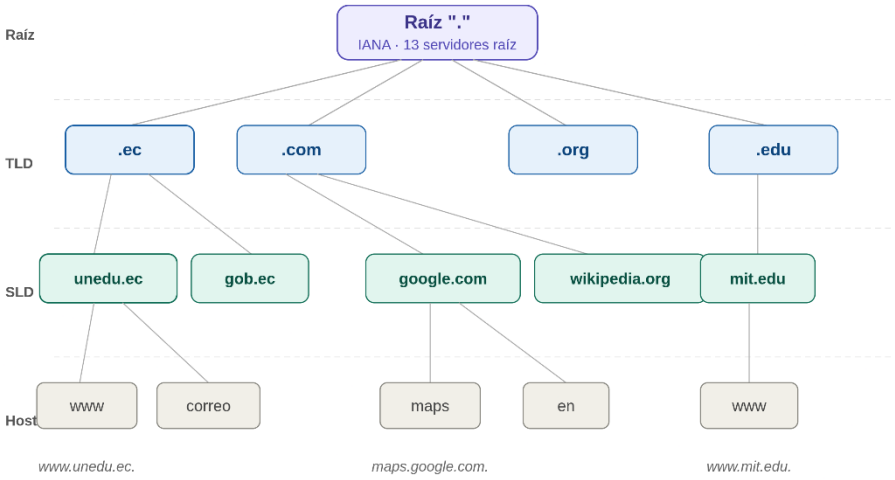
2.1.1. *Estructura Jerárquica del DNS*

El Sistema de Nombres de Dominio (DNS) se fundamenta en una estructura jerárquica como una pirámide, donde cada nodo representa un nivel dentro del espacio de nombres de dominio. El vértice absoluto de esta jerarquía es el dominio raíz, simbólicamente representado por un punto ("."). Inmediatamente subyacentes a este se encuentran los dominios de nivel superior (TLD), que a su vez actúan como nodos progenitores para los TLD de segundo nivel. Esta cadena descendente de autoridad se extiende progresivamente hasta alcanzar el nodo terminal, que designa un recurso específico. La secuencia completa de etiquetas que traza esta ruta desde el recurso hasta la raíz constituye un Nombre de Dominio Completamente Calificado (FQDN) como lo señala Laguna Ponce , (2023) en su tesis.

La administración del espacio de nombres se delega mediante zonas DNS. Una zona define una porción contigua del árbol de dominio sobre la cual un servidor DNS autoritativo ejerce control administrativo y responde con autoridad, la cúspide de esta estructura administrativa es la zona raíz ("."), responsable del dominio raíz. Esencialmente, todo FQDN culmina implícitamente con el punto raíz (p. ej., www.example.com.), aunque este último punto suele omitirse en el uso convencional. Esta notación completa (www.example.com.) es la expresión canónica del FQDN.

Figura 4

Jerarquía de Dominios DNS



Nota. Elaboración propia basada en Tanenbaum y Wetherall (2012) y Hoffman et al. (2019).

La Figura 4, representa el orden jerárquico que tiene el Sistema de Nombres de Dominio (DNS).

En el nivel superior se ubica el dominio raíz, que se representa por un punto ("."), que sirve como la base de todo el sistema que aunque no aparece visible en las direcciones web, este nivel es fundamental porque determina la ruta completa de los nombres de dominio (FQDN).

Se prosigue con la parte de abajo donde están los Dominios de Nivel Superior (TLDs), representados en la figura de López Padilla, (2022) por extensiones como .es, .com, .org y .net. Estas extensiones clasifican dominios por categorías, y actúan como puntos de delegación administrativa.

Un elemento que se muestra en la imagen de Tanenbaum et al., (2012) que es la conexión explícita de los dominios de segundo nivel (SLDs). Estos SLDs, como: inteco, gob, wikipedia e. internic, son administrados por entidades específicas y se ubican directamente bajo los TLDs. Como explica

Laguna Ponce (2023), "los SLDs identifican organizaciones únicas bajo los TLDs, formando la base identitaria de los dominios". Por ejemplo, .gob es un SLD bajo .es (formando gob.es), mientras que .wikipedia es un SLD bajo .org (wikipedia.org).

Como último paso, en el nivel inferior se encuentran los hosts o subdominios como www(web) y cert (certificaciones), que representan servicios específicos dentro de los dominios. Estos nodos terminales generan los nombres de dominio completamente calificados (FQDN[3]s) cuando se combinan con los niveles superiores, como www.wikipedia.org. o cert.internic.com., donde el punto final implícito referencia el dominio raíz según el modelo canónico.

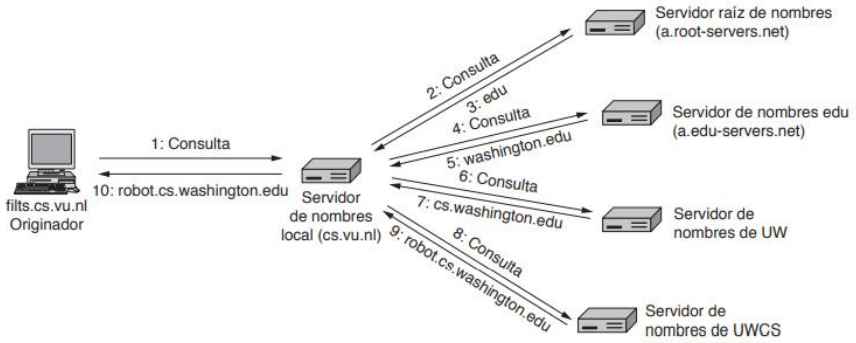
2.1.2. *Funcionamiento del DNS Tradicional*

Se conoce que el funcionamiento de DNS es un proceso que da resolución de nombres de sitios web y que sigue un modelo recursivo-iterativo. Para dar un ejemplo, cuando un cliente en este caso un navegador web solicita un dominio, un servidor recursivo normalmente operado por ISPs o servicios como Google DNS inicia la búsqueda jerárquica: Primero consulta los servidores raíz, que redirigen a los TLD (Top Level Domain) correspondientes, quienes finalmente dirigen al servidor autoritativo del dominio solicitado y este último devuelve el registro solicitado es decir la dirección IP y al traductor, lo almacena en caché temporalmente para optimizar consultas futuras. Este mecanismo de caché, aunque eficiente, representa un punto crítico de vulnerabilidad, pues carece de mecanismos para verificar la autenticidad de los datos recibidos. Todo el tráfico se transmite sin cifrado (protocolo UDP/53), (Mockapetris, 1987) exponiéndolo a interceptación.

[3] FQDN: Nombre de Dominio Completamente Calificado, es la dirección absoluta en DNS que termina con un punto (.), referenciando explícitamente la raíz. Ejemplo: www.gob.ec.

Figura 5

Pasos de resolución DNS.



Fuente: Tanenbaum, A. S., & Wetherall, D. J. (2012). Redes de computadoras (5a ed.). Pearson Educación.

La Figura 5 ilustra el funcionamiento jerárquico del DNS mediante un diagrama que detalla los pasos de resolución de nombres.

El proceso de resolución DNS descrito por Tanenbaum et al., (2012) se inicia cuando un host denominado “robot.cs.washington.edu” (originador) requiere que el resolutor un nombre de dominio en una dirección IP. El host envía una consulta recursiva a su servidor DNS local designado (cs.vu.nl), el cual opera como resolutor recursivo. Este servidor asume la responsabilidad completa de resolutor la consulta mediante consultas iterativas a través de la jerarquía DNS cuando no tiene la información solicitada en su caché local.

En el primer nivel de delegación, el servidor local contacta al servidor raíz (a. root-servers-net), un componente fundamental de la infraestructura DNS global que alberga información sobre los servidores autoritativos de los dominios de nivel superior (TLDs). El servidor raíz responde con una referencia al servidor TLD de .edu, el cual mantiene registros de los servidores autoritativos para los dominios de segundo nivel (SLDs) bajo su jurisdicción. Para el dominio

Washington.edu, el servidor TLD responde con la dirección del servidor de nombres autoritativo de la Universidad de Washington (que tiene una etiqueta como “Servidor de nombres UW” en la figura 5). Esta etapa muestra el principio de delegación administrativa, donde los TLDs gestionan la asignación de SLDs dentro de su espacio de nombres.

Al recibir esta referencia, al servidor local inicia una consulta al servidor autoritativo de washington.edu. Dado que el nombre completo solicitado (robot.cs.washington.edu) pertenece a un subdominio delegado (cs.washington.edu), este servidor responde con una delegación adicional, proporcionando la dirección del servidor de nombres específico para la subzona cs.washington.edu (denominado “Servidor de nombres UWCS”). Este nivel de delegación muestra cómo la autoridad se distribuye verticalmente en la jerarquía DNS, permitiendo una administración descentralizada.

Finalmente, el servidor local consulta al servidor autoritativo UWCS, que al ser responsable directo de la zona cs.washington.edu, responde con el registro de recurso solicitado (un registro AAAA correspondiente a la dirección IP de robot.cs.washington.edu). El servidor local almacena esta respuesta en su caché, respetando el tiempo de vida (TTL) especificado en el registro y la retorna al host. Este proceso completo, que involucra cuatro niveles de delegación (raíz, TLD, SDL, subdominio), ejemplifica los principios fundamentales del DNS que son: Distribución de Autoridad, resolución iterativa y almacenamiento jerárquico de información.

La eficiencia del sistema se basa en el mecanismo de caché, donde el servidor local (cs.vu.nl) retiene temporalmente las respuestas para acelerar consultas futuras, reduciendo así la carga sobre los servidores superiores. Simultáneamente, la estructura jerárquica garantiza la escalabilidad global pues cada entidad gestiona únicamente su zona de autoridad, mientras los servidores superiores mantienen solo referencias a las autoridades delegadas. Por lo que este diseño

permite resolver millones de consultas diarias manteniendo coherencia y rendimiento en un sistema esencial para la operación de Internet.

2.1.3. *Vulnerabilidades del DNS.*

La detección de fallas de seguridad relacionadas al Sistema de Nombres de Dominio se erige como un requisito ineludible para la puntuación de la eficacia de las mitigaciones orientadas a la preservación de su integridad operativa y disponibilidad. Dentro del espectro de amenazas que afectan a dicha estructura, merecen una especial consideración las siguientes:

Envenenamiento de la memoria de caché, este fenómeno conocido en la literatura como suplantación de identidad o spoofing, es un ataque donde un intruso introduce registros DNS falsos en la memoria caché de un servidor DNS recursivo (resolutor). Esto causa que las consultas legítimas de usuarios sean redirigidas a servidores maliciosos en lugar de sitios legítimos, facilitando el robo de datos, phishing y distribución de malware, su peligro se da porque este compromete a múltiples usuarios, hasta que el caché expire. Un ejemplo es el phishing bancario el cual se encuentra entre uno de los más comunes. Según el estudio de Alsad y Abu Al-Haija, (2023), este tipo de ataque se ha convertido en una amenaza persistente y sofisticada en el ámbito de la ciberseguridad. Aunque antes era considerado complejo y poco común, el avance de las herramientas y técnicas de ataque ha facilitado su ejecución incluso por actores maliciosos con recursos limitados.

Amplificación mediante denegación de servicio Distribuido, una variante particularmente peligrosa de este tipo de ataques es el denominado ataque de amplificación DNS, el cual explota la funcionalidad del sistema de nombres de dominio (DNS) para amplificar el volumen del tráfico malicioso enviado a un objetivo específico. Tal como lo explican S. Mekala, (2024), este tipo de ataque se caracteriza por aprovechar servidores DNS abiertos para reenviar grandes volúmenes de

respuestas DNS a una víctima, generando un colapso en su red o en sus servicios. Por ejemplo, una consulta de 60 bytes puede generar respuestas de hasta 3.200 bytes usando registros tipo ANY o EDNS, logrando coeficientes de amplificación de hasta 54:1. Esto satura el ancho de banda o recursos de la víctima con tráfico legítimo originado en infraestructura DNS comprometida.

Apropiación indebida de dominios, la usurpación de la titularidad o de la configuración resolutiva de un dominio implica la modificación ilegitima de las respuestas del sistema, conduciendo a los usuarios hacia destinos espurios. De dicha alteración se derivan afectaciones concurrentes sobre la confidencialidad de las comunicaciones y sobre la garantía de autenticidad de los contenidos recibidos, al quedar el proceso de resolución subvertido por la intervención de un tercero no autorizado.

Según Houser et al., (2021), el secuestro de DNS se ha convertido en una práctica frecuente en diversas partes del mundo, muchas veces llevada a cabo no solo por actores maliciosos individuales, sino incluso por proveedores de servicios de Internet (ISP) o autoridades gubernamentales. Esta forma de manipulación puede realizarse a través de múltiples mecanismos, como la modificación de configuraciones en resolutores DNS, la interceptación del tráfico en tránsito o el uso de resolutores alternativos que devuelven respuestas falsas.

2.2 Extensión de Seguridad DNSSEC.

DNSSEC (Domain Name System Security Extensions) es un conjunto de extensiones que protegen la autenticidad e integridad de los datos DNS mediante criptografía asimétrica. Su objetivo principal es permitir que un resolutor cliente DNS con capacidad de validación verifique tres aspectos críticos; Autenticidad del origen, es quién confirma que los datos recibidos provienen exclusivamente de la zona autoritativa legítima y no de alguien que intenta suplantarlo, como lo señala Ron Aitchison, (2011) ; la integridad de los datos, que garantiza que la información

no ha sido adulterada durante la transmisión, protegiendo el contenido, Prueba de inexistencia que valida una respuesta negativa (NXDOMAIN[4]) es genuina, demostrando criptográficamente que el registro solicitado no existe en la zona. Por lo que *"DNSSEC no cifra datos ni protege la privacidad; su función es garantizar autenticidad e integridad mediante criptografía de clave pública y firmas digitales en registros especializados"* (Aitchison, 2011, p. 247).

2.2.1. *Componentes de la extensión DNSSEC*

La implementación de DNSSEC se fundamenta en cuatro registros esenciales que establecen una cadena criptográfica jerárquica. El registro RRSIG (Resource Record Signature) contiene firmas digitales generadas con claves privadas ZSK (Zone Signing Key) que cubren conjuntos completos de registros (RRsets), garantizando la integridad y autenticidad de los datos. Complementariamente, el registro DNSKEY almacena las claves públicas en dos variantes críticas: las ZSK para firmar datos de la zona y las KSK (Key Signing Key) que establecen jerarquías criptográficas al firmar otras claves.

Para crear enlaces de confianza entre zonas, el registro DS (Delegation Signer) almacena en la zona padre una huella criptográfica (hash SHA-256/SHA-384) de la KSK de la subzona, formando eslabones verificables y finalmente, el NSEC3 proporciona prueba de inexistencia de nombres mediante técnicas de hashing con sal criptográfica, evitando la enumeración de zonas mientras demuestra la ausencia válida de registros.

Por ejemplo si al realizar una consulta para "github.com", se activa un proceso de validación multicapa. El servidor recursivo recibe el RRset A (direcciones IP) junto con su

[4] **NXDOMAIN:** Significa "dominio no existente". Es un mensaje que indica que la dirección web o el nombre de la página que intentaste buscar no se encontró en internet. Si ves esto con DNSSEC, significa que el mensaje de "no existe" es auténtico y seguro.

correspondiente RRSIG y la DNSKEY pública de la zona. Inmediatamente verifica la firma RRSIG utilizando la DNSKEY pública, confirmando que los datos no han sido alterados y proceden de la fuente legítima. Para autenticar la propia DNSKEY, el servidor recursivo solicita a la zona padre (.com) el registro DS asociado a github.com, que contiene el hash criptográfico de la KSK. Al generar localmente el hash de la KSK recibida y compararlo con el valor DS almacenado en la zona superior, se valida el eslabón de confianza.

Este proceso se repite recursivamente hasta la raíz DNSSEC, creando una cadena de autenticación inquebrantable, pero en caso de respuestas negativas (NXDOMAIN), el NSEC3 demuestra mediante secuencias hasheadas que el nombre solicitado no existe en la zona, completando así el modelo de seguridad integral que protege tanto la existencia como la inexistencia de registros.

La coherencia criptográfica del sistema emerge de la interdependencia de estos componentes: las RRSIG protegen datos específicos mediante firmas efímeras, mientras las DNSKEY establecen identidades criptográficas duraderas. Los DS convierten delegaciones DNS en eslabones verificables, y los NSEC3 cierran el modelo asegurando que las respuestas negativas sean tan confiables como las positivas. Esta arquitectura permite que resolutores validantes autentiquen cada respuesta mediante operaciones locales sin requerir canales seguros, implementando lo que Aitchison (2011) describe como "un modelo de confianza descentralizado donde cada componente refuerza la autenticidad del conjunto mediante pruebas criptográficas independientes" (RFC 4034).

2.2.2. *Cadena de Confianza y Validación.*

El proceso de validación DNSSEC comienza en resolutor[5], el cual parte desde un ancla de confianza preconfigurada correspondiente a la zona raíz del DNS. Esta ancla es una clave pública que puede ser instalada manualmente o actualizada automáticamente mediante el mecanismo definido en la RFC 5011. A partir de esta clave, el resolutor valida la firma del registro DS[6] perteneciente a la zona de nivel superior (por ejemplo, .com) utilizando la clave raíz. Luego, con el registro DS validado, se verifica la clave pública DNSKEY[7] de un dominio específico (como ejemplo.com), asegurando que esta clave fue debidamente delegada por su zona superior. Una vez establecida esta relación de confianza, el resolutor puede comprobar la firma RRSIG[8] de los registros finales como los tipos A o MX utilizando la clave de firma de zona (ZSK) correspondiente.

Este proceso encadenado garantiza que cada nivel ha sido autorizado por su antecesor en la jerarquía. Es importante destacar que cualquier ruptura en esta cadena de validación, ya sea por una clave incorrecta, un registro DS inválido o una firma digital alterada, resulta en el rechazo de toda la respuesta DNS. En tales casos, el servidor devuelve un error SERVFAIL[7], indicando que la validación de seguridad ha fallado. Por lo tanto, el correcto funcionamiento del sistema depende totalmente de la integridad y disponibilidad de cada uno de los eslabones de la cadena de confianza, haciendo del trust anchor un elemento esencial en la seguridad del ecosistema DNS.

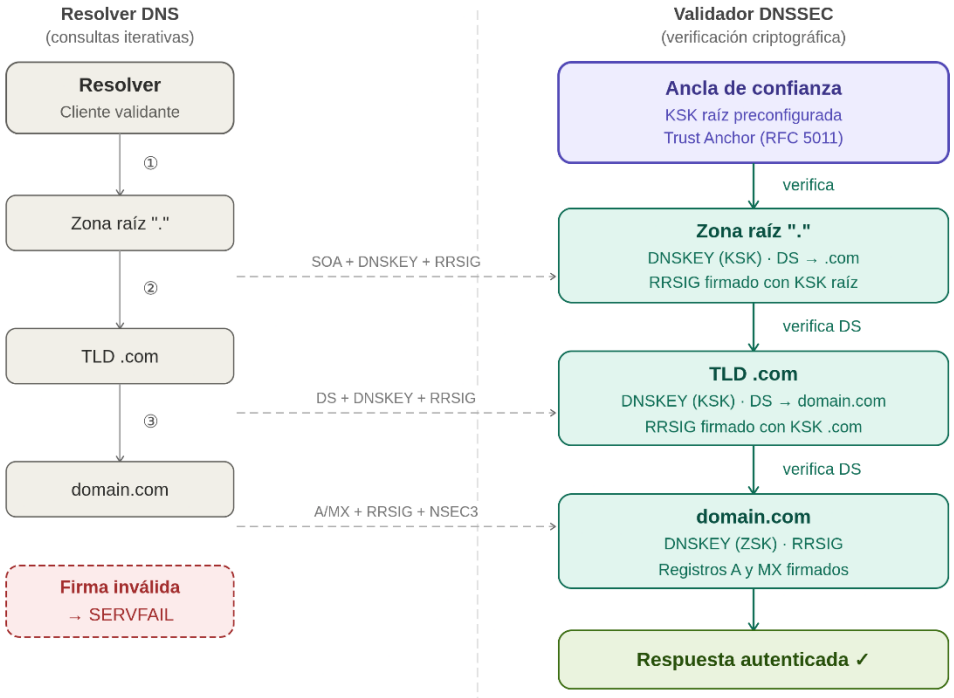
[5] Resolutor: Es el servidor DNS que se encarga de buscar las direcciones IP correspondientes a los nombres de dominio que escribimos en el navegador.

[6] Registro DS: Es un tipo especial de registro DNS que actúa como un "vínculo firmado" entre un dominio superior (como .com) y un dominio hijo. Sirve para confirmar que el dominio hijo está autorizado por el dominio padre y que su clave pública es confiable.

[7] SERVFAIL: Es un mensaje de error que devuelve el servidor DNS cuando algo falla durante el proceso de validación, como una firma digital inválida o una clave faltante.

Figura 6

Cadena de Confianza y Validación de DNSSEC



El ancla de confianza (trust anchor) en DNSSEC es un elemento crítico que establece el punto de partida criptográfico para la validación de la cadena de confianza. Consiste en una o más claves públicas previamente verificadas y confiables, asociadas a una zona DNS específica (generalmente la zona raíz o un TLD), que se almacenan localmente en el resolutor DNS. Estas claves actúan como referencia inmutable: cuando un resolutor recibe una respuesta firmada digitalmente (por ejemplo, para el dominio ejemplo.gob.mx), inicia el proceso de validación comparando la firma recibida con la clave pública del nivel superior jerárquico, escalando recursivamente hasta llegar al ancla de confianza. Si todas las firmas a lo largo de esta jerarquía son verificadas exitosamente contra las claves almacenadas en el ancla, los datos se consideran auténticos e íntegros.

La singularidad del ancla se refiere a que no depende de validaciones externas en tiempo real: Es instalada manualmente o mediante canales seguros (como paquetes de sistemas operativos o actualizaciones firmadas por entidades certificadoras), convirtiéndose en la "raíz de confianza" del sistema. Por ejemplo, la clave pública de la zona raíz DNS (operada por IANA) es distribuida como ancla global, y resolutores como Unbound o BIND la incluyen por defecto en su configuración. Si esta clave raíz fuera comprometida, toda la cadena de confianza DNSSEC colapsaría, lo que subraya su papel como único punto irreemplazable que debe protegerse mediante almacenamiento offline y rotaciones controladas bajo protocolos como el RFC 5011 (Actualización Automática de Anclas de Confianza).

El Ancla de Confianza es la clave pública raíz (KSK) preconfigurada en el validador DNSSEC. Actúa como la base incuestionable de toda la verificación, almacenada como *Trust Anchor*.

La Zona Raíz Contiene su clave KSK (firma otras claves) en un registro DNSKEY y genera el registro DS para el TLD (.com): un resumen criptográfico de la clave KSK del TLD, vinculando ambas zonas.

La Zona TLD (.com) recibe el DS de la raíz y lo usa para autenticar su propia clave KSK generando un nuevo DS para el dominio final (*domain.com*), extendiendo la cadena.

El Dominio Final (domain.com), usa el DS del TLD para verificar su clave KSK y esta KSK firma los datos DNS locales (como registros A, MX) mediante RRSIG, garantizando su integridad. IP eficiente, (s/f). Como resume la RFC 4033, *"DNSSEC construye confianza mediante una jerarquía criptográfica donde cada zona autentica la clave de la zona inferior"* .

2.2.3. *RFC 6841: Declaraciones de Políticas y Prácticas.*

La gobernanza y operación segura del sistema DNSSEC se rige por un conjunto de documentos normativos fundamentales que garantizan la transparencia, la responsabilidad y la interoperabilidad entre entidades operadoras. Entre estos se destacan la Declaración de Políticas de DNSSEC (DNSSEC Policy, DP) y la Declaración de Prácticas de DNSSEC (DNSSEC Practice Statement, DPS). La Declaración de Políticas define los principios generales y los requisitos de seguridad que una zona DNS debe cumplir para operar bajo DNSSEC, incluyendo aspectos como los algoritmos criptográficos autorizados, el nivel de protección requerido para las claves privadas, la duración de validez de las firmas digitales, y los estándares internacionales aplicables en materia de gestión de claves.

Por su parte, la Declaración de Prácticas detalla de manera precisa cómo se implementan estas políticas en la operación diaria, incluyendo la frecuencia de rotación de claves criptográficas (KSK y ZSK), las medidas de seguridad física y lógica aplicadas a los módulos de seguridad hardware (HSM), los procedimientos de respaldo, auditoría y monitoreo continuo, así como los protocolos de recuperación ante desastres o incidentes de seguridad. Estas declaraciones son redactadas y publicadas por los operadores de zonas como ICANN, en el caso del dominio raíz y dominios de primer nivel como .com, o EURid para .eu, y su disponibilidad pública responde al principio de transparencia operativa exigido por el ecosistema global del DNS. La normativa que orienta la estructura y el contenido de estos documentos está descrita en la RFC 6841, la cual proporciona un marco estandarizado para comunicar de forma clara y verificable las políticas y prácticas de seguridad de cada zona firmada con DNSSEC.

2.3 Privacidad en DNS: Desafíos y Soluciones.

2.3.1. *Riesgo de Exposición de Datos.*

El DNS tradicional opera mediante consultas en texto claro, exponiendo metadatos sensibles como dominios visitados, patrones de navegación e identidad de usuarios. Esta vulnerabilidad permite a proveedores de Internet (ISPs), gobiernos y actores maliciosos construir perfiles detallados de comportamiento mediante sniffing pasivo, violando el principio de confidencialidad. Estudios como el de ICANN (2020) revelan que el 78% de las consultas DNS globales carecen de cifrado, facilitando la vigilancia masiva. Un caso emblemático ocurrió en 2017 cuando proveedores estadounidenses vendieron historiales de navegación a anunciantes, evidenciando cómo el modelo jerárquico centralizado convierte a los resolutres en puntos únicos de fuga de datos.

En la tabla 1 se resumen las vulnerabilidades más significativas del protocolo DNS y se detalla el mecanismo criptográfico específico que utiliza DNSSEC para neutralizar cada amenaza.

DNSSEC proporciona contramedidas específicas para las amenazas más críticas y estas van desde la validación de firmas digitales (RRSIG) para prevenir el envenenamiento de caché y la suplantación de identidad hasta el uso de registros NSEC3 para impedir la enumeración de zonas.

Tabla 1

Tipos de ataques y Mitigación mediante DNSSEC

Ataque	Tipo de Ataque	Afectación Principal	Mecanismo de Mitigación DNSSEC
Envenenamiento de Caché	Inyección de Datos Maliciosos	Inyección de registros falsos en la caché del SDRV.	El SDRV descarta cualquier respuesta cuya firma digital (RRSIG) no pueda ser

		Redirección a sitios maliciosos.	verificada con la ZSK pública auténtica de la zona.
Suplantación de Identidad	Intercepción y Manipulación de Datos	Alteración de los registros DNS en tránsito entre servidores.	La validación de la Cadena de Confianza garantiza que cualquier modificación en el RRset invalida la RRSIG asociada, forzando la respuesta SERVFAIL.
Ataque Birthday	Predicción de ID y Puerto	Inyección de respuesta falsificada antes que el SNA legítimo, explotando la debilidad del TXID.	La seguridad se basa en la Criptografía Asimétrica (claves), no en la aleatoriedad del ID de transacción. La respuesta debe tener una firma válida.
Enumeración de Zona	Recolección de Información	Obtención de todos los nombres de dominio de una zona, facilitando ataques dirigidos.	El uso de registros NSEC3 reemplaza los nombres de dominio en texto plano por valores hash, impidiendo la enumeración directa de la zona.
Falsificación de Inexistencia	Bloqueo de Acceso (DoS)	Inyección de una respuesta NXDOMAIN falsa para bloquear el acceso a un dominio legítimo.	Las respuestas NXDOMAIN deben estar acompañadas de registros NSEC/NSEC3 firmados que prueban criptográficamente la inexistencia del dominio.

2.3.2. *Protocolos de Cifrado (DoH/DoT).*

En paralelo al enfoque de validación de integridad y autenticidad que ofrece DNSSEC, han surgido otras tecnologías orientadas a proteger la confidencialidad de las consultas DNS durante su transmisión. Entre ellas destacan los protocolos DNS over HTTPS (DoH) y DNS over TLS

(DoT), los cuales implementan mecanismos de cifrado de extremo a extremo para evitar que actores intermedios (como proveedores de servicios de Internet, redes públicas o atacantes con acceso al canal de comunicación) intercepten, monitoreen o manipulen las solicitudes DNS. A diferencia de DNSSEC, que protege la autenticidad de las respuestas mediante firmas digitales, pero no cifra el contenido, DoH y DoT se centran en ocultar las consultas y respuestas entre el cliente y el resolutor, previniendo técnicas como el secuestro de DNS o la vigilancia del tráfico por parte de terceros.

En el contexto ecuatoriano, donde los usuarios suelen utilizar redes compartidas en universidades, espacios públicos y proveedores locales con resolutores que no siempre implementan políticas de seguridad robustas, la adopción de estos protocolos representa una mejora significativa en la protección de la privacidad digital. Algunos operadores de telecomunicaciones en Ecuador aún no implementan DoH/DoT de forma nativa, lo que obliga a los usuarios a depender de resolutores externos (como los de Cloudflare o Google) para beneficiarse de estas tecnologías. No obstante, su implementación progresiva en el país podría contribuir a fortalecer el ecosistema DNS nacional, complementando la seguridad estructural provista por DNSSEC con la confidencialidad en tiempo real que brindan estos protocolos cifrados. Esta combinación resulta especialmente relevante frente a escenarios de censura, interferencia o ataques dirigidos que pueden comprometer tanto la integridad como la privacidad del sistema de resolución de nombres en entornos locales.

2.3.3. *Marco Regulatorio.*

La implementación de mecanismos de seguridad como DNSSEC y los protocolos de cifrados asociados al sistema de nombres de dominio (DNS) no solo representan un desafío técnico, sino que también requieren el respaldo de un marco regulatorio adecuado que promueva

su adopción y operación segura; en el caso de Ecuador, el entorno normativo en materia de telecomunicaciones y seguridad de la información está regido por organismos como el Ministerio de Telecomunicaciones y de la Sociedad de la Información (MINTEL), la Agencia de Regulación y Control de las Telecomunicaciones (ARCOTEL) y el Centro de Respuesta a Incidentes Informáticos del Ecuador (ECU-CERT). Aunque actualmente no existe una legislación específica que obligue la implementación de DNSSEC en dominios nacionales .ec, sí existen disposiciones generales contenidas en la Ley Orgánica de Telecomunicaciones y la Ley Orgánica de Protección de Datos Personales, que abren el camino hacia políticas más estrictas de seguridad, integridad y confidencialidad en las redes.

En la Ley de Protección de Datos Personales (2021) establece que los responsables del tratamiento de datos deben aplicar medidas técnicas y organizativas apropiadas para garantizar la seguridad de la información, lo que incluye proteger las comunicaciones DNS contra manipulaciones o interceptaciones, para lo cual estas tecnologías como DNSSEC pueden considerarse como parte de las mejores prácticas para cumplir con los principios de integridad y disponibilidad de los datos, en especial en instituciones públicas, entidades financieras y proveedores de servicios digitales. A nivel de dominios, el operador de la zona .ec, actualmente gestionado por NIC.EC, ofrece soporte técnico para la firma DNSSEC desde 2017; sin embargo, la activación depende de cada registrante, y no está regulada de forma obligatoria. Esto resalta la necesidad de contar con normativas más específicas que impulsen la adopción de DNSSEC como estándar de seguridad nacional, similar a los lineamientos establecidos por entidades como ICANN o los marcos del Foro de Cooperación en Ciberseguridad de América Latina y el Caribe (LAC4).

3. Capítulo III: Diseño e implementación

Este capítulo describe el diseño, implementación y configuración del entorno de simulación que permite evaluar la infraestructura DNS tradicional y la extensión DNSSEC frente a ataques de Flooding y Spoofing. La metodología experimental se estructura en cuatro escenarios (E1 a E4) que combinan la activación o desactivación de DNSSEC con la ausencia o presencia de una carga adversaria, lo cual permite establecer comparaciones cuantitativas objetivas sobre el rendimiento, la disponibilidad y la efectividad de las medidas de protección. Los cuatro escenarios operan bajo condiciones de carga controlada y están instrumentados para la captura de métricas que evalúan el impacto del ataque en términos de integridad de datos, latencia de respuesta y consumo de recursos computacionales.

3.1 Enfoque y Metodología de Investigación

La presente investigación adopta un enfoque cuantitativo con diseño experimental controlado, orientado a evaluar la efectividad del protocolo DNSSEC como mecanismo de protección frente a ataques de Flooding y Spoofing . En este diseño, la variable independiente es la activación o desactivación de DNSSEC en la infraestructura DNS, mientras que las variables dependientes corresponden a las métricas de seguridad y de rendimiento (latencia, throughput, uso de recursos y tamaño de los paquetes DNS). El entorno experimental se implementa en un laboratorio virtualizado mediante GNS3 con contenedores Docker, lo que permite controlar la totalidad de las variables del entorno y garantizar la reproducibilidad de los experimentos, aislando el efecto de DNSSEC de factores que distorsionarían las mediciones en un entorno de producción, como la variabilidad de la latencia de Internet, la carga de tráfico externo y las políticas de seguridad perimetral.

En primer lugar, se estudian las estructuras y características fundamentales del DNS tradicional su arquitectura jerárquica y las vulnerabilidades de su caché y de DNSSEC las firmas RRSIG, las claves DNSKEY y la cadena de confianza basada en registros DS, analizando sus diferencias en seguridad y funcionamiento. Este fundamento conceptual orienta el diseño de la topología de tres zonas (LAN, tránsito y DMZ) con direccionamiento de doble pila IPv4.

Sobre esa base, se implementa el entorno de simulación desplegando los nodos sobre contenedores Docker con BIND 9.20.23. Se configura primero la resolución DNS tradicional como línea base y, posteriormente, se habilita DNSSEC mediante la firma estática de la zona con dnssec-signzone, empleando el algoritmo ECDSAP256SHA256 (algoritmo 13) [8] y NSEC3[9], junto con la generación del par de claves KSK[10] y ZSK[11] y el establecimiento de los anclajes de confianza en el servidor recursivo. Los parámetros relevantes, como valores de TTL y volumen de consultas se estandarizan para garantizar la comparabilidad entre escenarios. Este entorno permite, además, emular los ataques de envenenamiento de caché y generar cargas de tráfico controladas para la medición cuantitativa, dando cumplimiento al segundo objetivo específico.

Siguiendo la línea, se ejecutan los cuatro escenarios experimentales que estructuran la evaluación: E1, DNS tradicional; E2, DNS con DNSSEC en condiciones normales; E3, DNS sin DNSSEC bajo carga adversaria; y E4, DNS con DNSSEC bajo carga adversaria. Cada escenario de ataque se subdivide en una componente cuantitativa (medición de rendimiento bajo la carga adversaria) y una componente cualitativa (verificación del comportamiento de validación ante una

[8] **ECDSAP256SHA256:** Algoritmo de Firma Digital de Curva Elíptica. Es un estándar criptográfico altamente eficiente utilizado en DNSSEC para generar firmas digitales cortas y rápidas de validar.

[9] **NSEC3:** Registro de seguridad que prueba criptográficamente que un subdominio no existe

[10] **KSK (Key Signing Key):** Clave criptográfica principal y raíz de confianza de una zona DNSSEC. Su única función es firmar a la clave secundaria (ZSK).

[11] **ZSK (Zone Signing Key):** Clave criptográfica secundaria en DNSSEC que se encarga de firmar digitalmente los registros de datos del día a día.

respuesta forjada). Durante su ejecución se genera tráfico legítimo y de fondo para simular el uso concurrente de múltiples usuarios y se lanzan los ataques mediante los scripts desarrollados.

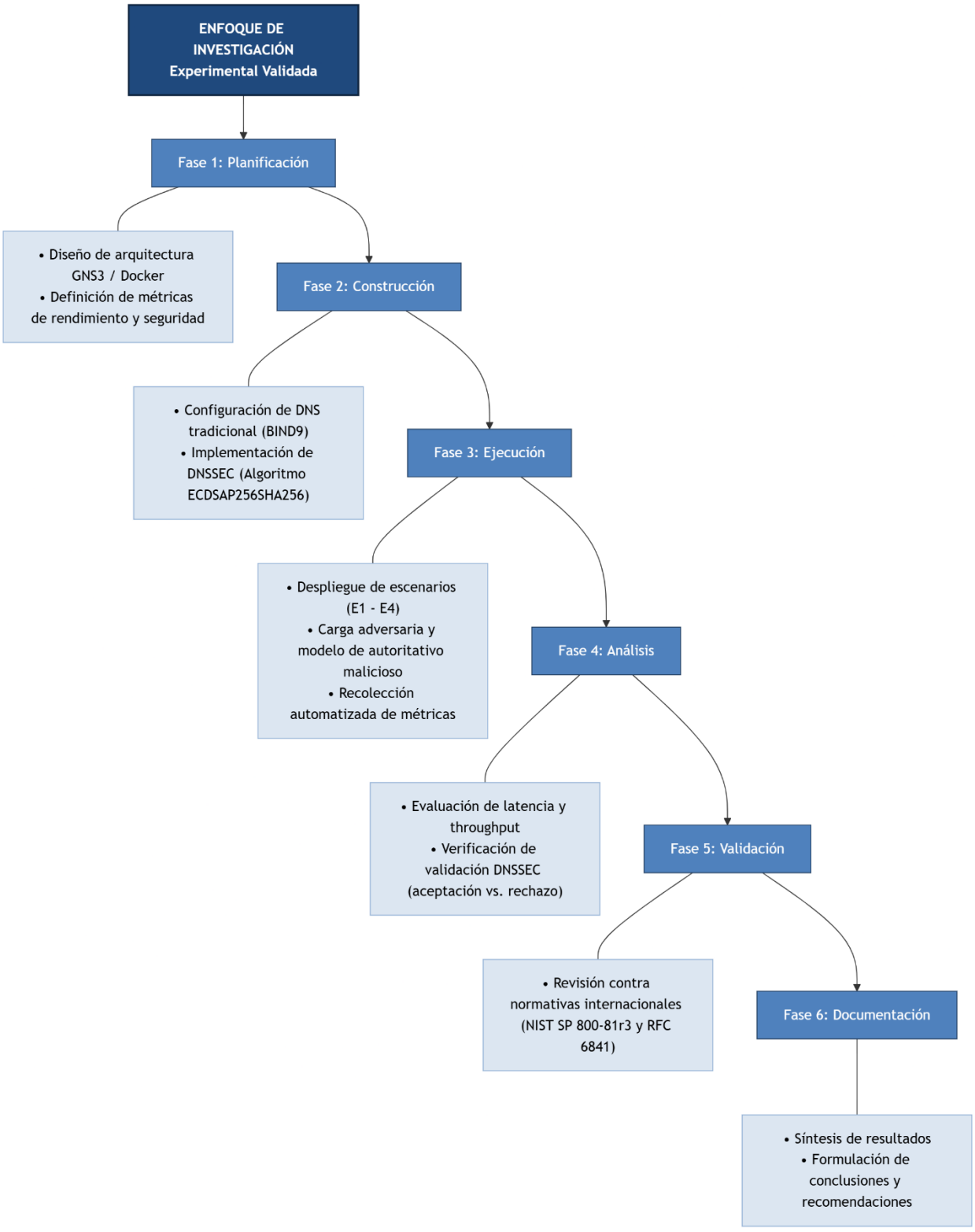
El diseño responde a una matriz de dos variables independientes, la activación de DNSSEC (desactivada/activada) y la presencia de carga adversaria (ausente/presente). El cruce de ambas genera los 4 escenarios: E1 (sin DNSSEC, sin ataque) como línea base; E2 (con DNSSEC, sin ataque) para aislar el costo de la validación; E3 (sin DNSSEC, bajo ataque) y E4 (con DNSSEC, bajo Ataque) para evaluar el comportamiento bajo condiciones adversarias. Esta estructura permite tres tipos de comparación: el overhead de DNSSEC (E2 frente a E1); el efecto del ataque, evaluado tanto sin DNSSEC (E3 frente a E1) como con DNSSEC (E4 frente a E2); y el costo de DNSSEC bajo ataque (E4 frente a E3).

En paralelo, se comparan cuantitativamente ambos sistemas capturando de forma sistemática la latencia y el throughput con dnsperf, bajo distintas cargas de consultas, verificando la resolución con dig, inspeccionando los paquetes con los analizadores de tráfico (Wireshark/tshark) y registrando el uso de recursos de los contenedores, tanto en condiciones normales como bajo escenarios de ataque y carga variable, lo que responde al tercer objetivo específico.

A partir de ello se interpretan los resultados con el fin de determinar el impacto de DNSSEC en el entorno simulado, identificando fortalezas, debilidades y compromisos entre seguridad y rendimiento. Estos hallazgos se contrastan con los marcos normativos internacionales de la guía NIST SP 800-81r3 y las especificaciones de las RFC 4033-4035, 5155 y 6841 para asegurar su coherencia con las buenas prácticas establecidas, cumpliendo así el cuarto objetivo específico y consolidando la base sobre la cual se elabora el informe técnico y se formulan las recomendaciones prácticas de la investigación.

Figura 7

Fases de la Metodología implementada



Como se detalla la Figura 7, el proceso inicia con una fase de Planificación y Construcción, donde se despliega la topología lógica y se aíslan los servicios DNS. Posteriormente, la fase de Ejecución se encarga de inyectar la carga adversaria y de verificar el comportamiento de validación en los cuatro escenarios definidos. Siguiendo las fases de Análisis y Validación se asegura empíricamente que los resultados obtenidos sean contrastados con los estándares internacionales, específicamente las directrices del NIST SP 800-81r3 y el RFC 6841, garantizando la objetividad técnica de las conclusiones. La última fase corresponde a la documentación, que sintetiza los resultados y da lugar a la formulación de conclusiones y recomendaciones.

3.2 Requerimientos.

Esta sección detalla los factores requeridos para el cumplimiento de los objetivos del proyecto utilizando tecnologías de virtualización. El propósito es asegurar que la comparación entre DNS y DNSSEC, operen en un entorno que emule fielmente el comportamiento de servidores independientes.

La ejecución de este apartado se desarrolla conforme a la norma ISO/IEC/IEEE 29148:2018.

3.2.1. *Requerimientos de Stakeholders (StSR)*

Los stakeholders representan a las partes interesadas que intervienen directa o indirectamente en el desarrollo del sistema. La definición de los requerimientos comienza con la identificación de los actores clave que interactúan directa o indirectamente con el proyecto de investigación que constituye la etapa fundamental para alinear los objetivos técnicos de la simulación. La identificación de estos roles permite establecer un canal de validación para los resultados obtenidos en la comparativa de seguridad y rendimiento.

En la Tabla 2 se detallan los actores con autoridad, quienes poseen la autoridad para aprobar los entregables y verificar el cumplimiento de los objetivos.

Tabla 2

Requerimientos de Stakeholders

Nro.	Stakeholders	Rol
StSR1	Ing. Mauricio Domínguez	Tutor de Tesis
StSR2	Ing. Fabián Cuzme	Asesor de Tesis
StSR3	Brian Steve Rea	Estudiante

3.2.2. *Requerimientos del Sistema (SySR)*

Una vez identificados los actores, se establece las capacidades técnicas que el sistema debe poseer para soportar la simulación dado que el estudio se centra en un análisis comparativo, el sistema debe ser capaz de replicar fielmente un entorno de producción, pero bajo condiciones controladas de laboratorio.

Los requerimientos del sistema, presentados en la Tabla 3, se han categorizado en funcionales, de rendimiento y de seguridad, priorizando aquellos elementos para la ejecución de ataques de envenenamiento de caché y la validación de firmas criptográficas.

Tabla 3

Requerimientos del Sistema (SySR)

Requerimientos de la Interfaz y Operación.		
ID	**Descripción**	**Prioridad (Alta/Media/Baja)**

SySR1	El sistema debe simular una infraestructura de red mediante orquestación de contenedores Docker integrados en GNS3, permitiendo el despliegue de nodos DNS, clientes, atacantes y servicios web en un entorno que comparte el kernel del host para optimizar el uso de recursos.	Alta
SySR2	Capacidad para automatizar el despliegue y reinicio del escenario mediante "Infraestructura como Código" (Docker Compose) para restablecer el entorno a su estado base en menos de 60 segundos.	Alta
SySR3	El sistema debe gestionar privilegios de red avanzados (Capabilities NET_ADMIN/NET_RAW) en los contenedores para permitir la inyección de paquetes raw y spoofing de IP necesarios para el ataque.	Alta
Requerimientos de Performance (Rendimiento)		
ID	**Descripción**	**Prioridad (Alta/Media/Baja)**
SyPR4	El sistema debe soportar la ejecución simultánea de al menos 14 nodos (9 contenedores Docker, 2 routers emulados y 3 switches virtuales) manteniendo un consumo de RAM inferior a 8 GB mediante el uso de contenedores que comparten el kernel del host.	Alta
SyPR5	La arquitectura debe permitir la generación de al menos 200 consultas DNS por escenario experimental, distribuidas entre 30 dominios diferentes, para garantizar significancia estadística en las mediciones de latencia (promedio, desviación estándar, valores mínimo y máximo).	Alta

SyPR6	El entorno debe ser capaz de generar y procesar un flujo de tráfico DNS combinado de hasta 300 consultas por segundo para evaluar el throughput bajo condiciones de carga progresiva en los servidores de resolución.	Media
SyPR7	El sistema debe completar el arranque y la convergencia de enrutamiento de toda la topología en menos de 60 segundos para optimizar los ciclos de prueba y error.	Media
Requerimientos de Seguridad		
	Descripción	**Prioridad (Alta/Media/Baja)**
SySR8	Los contenedores designados como atacantes deben contar con capacidades extendidas (NET_ADMIN, NET_RAW) para permitir la construcción y envío de paquetes manipulados (IP Spoofing) necesarios para el envenenamiento de caché.	Alta
SySR9	Los nodos de la infraestructura (basados en Debian con BIND 9.20.23) deben integrar bibliotecas criptográficas optimizadas para realizar la validación completa de la cadena de confianza DNSSEC (RRSIG, DNSKEY, DS) utilizando el algoritmo ECDSAP256SHA256.	Alta
SySR10	El sistema debe segregar el tráfico de gestión del tráfico de ataque mediante redes virtuales diferenciadas, asegurando que la captura de datos (pcap) no se contamine con ruido del sistema anfitrión.	Media

3.2.3. *Requerimientos de Software (SRS)*

Para materializar los requerimientos del sistema descritos anteriormente, es necesario seleccionar un conjunto de herramientas de software que sean compatibles entre sí y que permitan la orquestación de la topología en un entorno local. La elección del software se basa en criterios de código abierto, flexibilidad y soporte para estándares de internet actuales.

La Tabla 4 describe la configuración de software base, destacando el uso de sistemas Linux para la gestión nativa de los protocolos de red.

Tabla 4

Requerimientos de Software (SRS)

Requerimientos de Software (SRS)		
ID	**Descripción**	**Prioridad (Alta/Media/Baja)**
SySR1	EndeavourOS (distribución basada en Arch Linux) con Docker Engine 29.5.1 y soporte de virtualización habilitado para la integración con GNS3 3.0.6.	Alta
SySR2	Uso de Docker Engine orquestado dentro de GNS3 para la ejecución de contenedores ligeros que comparten el kernel del host (reemplazando QEMU/KVM).	Alta
SySR3	Imágenes personalizadas basadas en Python:3.11-slim (Debian 12) para los servidores DNS, clientes y atacantes, construidas mediante Dockerfiles que incluyen BIND 9.20.23 y las herramientas necesarias para cada rol. El nodo atacante incluye Scapy 2.7.0 para la construcción de paquetes DNS falsificados.	Alta

SySR4	BIND9 desplegado e inyectado automáticamente en los contenedores mediante volúmenes o instrucciones COPY en el Dockerfile, eliminando la configuración manual.	Alta
SySR5	Entorno Python 3.11 con la biblioteca Scapy 2.7.0 integrada en los contenedores atacantes para la ejecución de scripts de Flooding y DNS Spoofing (dns_falso.py)	Alta
SySR6	Scripts automatizados de medición basados en dig (BIND9 utils) con exportación de resultados a archivos CSV para su análisis estadístico. El contenedor Analizador permita la captura de tráfico DNS para verificación de las consultas y respuestas.	Alta

3.2.4. *Especificación de Requerimientos del Modelo de Análisis (MRS)*

Más allá del software y hardware, el diseño del experimento debe regirse por reglas que aseguren la validez científica de los datos. El modelo de análisis define las condiciones bajo las cuales se ejecutarán las pruebas de estrés y seguridad, garantizando que los resultados observados (éxito del ataque o incremento de latencia) sean consecuencia directa de las variables estudiadas y no de errores en el entorno de pruebas.

En la Tabla 5 se especifican las restricciones y condiciones operativas del modelo de simulación.

Tabla 5

Requerimientos del Modelo de Análisis

ID	Descripción	Prioridad
MRS1	El estado base de los servidores debe ser inmutable, definido por código (Dockerfiles), garantizando que cada reinicio restaure la	Alta

	configuración exacta de "producción" sin residuos de ataques anteriores.	
MRS2	El modelo debe utilizar el stack TCP/IP nativo del Kernel Linux del host (vía Network Namespaces) para procesar el tráfico real, evitando la sobrecarga de la emulación de hardware de las VMs.	Alta
MRS3	Se deben aplicar límites estrictos de CPU y RAM (Control Groups) a cada contenedor para garantizar que la degradación del servicio sea causada exclusivamente por el tráfico del ataque y no por el agotamiento de recursos del host físico.	Alta
MRS4	Capacidad para restaurar todo el escenario (13 nodos) a su estado base mediante scripts de configuración, permitiendo múltiples iteraciones experimentales para garantizar la reproducibilidad de los resultados.	Media

3.2.5. *Especificación de Requerimientos de Arquitectura (SyRA)*

La integración de todos los componentes anteriores se cristaliza en una arquitectura de red lógica. Esta arquitectura debe reflejar la estructura jerárquica del Sistema de Nombres de Dominio (DNS) real, permitiendo simular el flujo de una consulta desde un cliente en una LAN hasta los servidores raíz.

La Tabla 6 establece las directrices topológicas que guiarán el diseño del escenario en el simulador GNS3.

Tabla 6

Requerimientos de Arquitectura

ID	Descripción	Prioridad (Alta/Media/Baja)
SyRA1	La arquitectura debe integrar contenedores Docker (para servicios DNS, clientes y atacantes) con routers Cisco C7200 emulados con Dynamips dentro de GNS3, permitiendo una topología realista con consumo optimizado de recursos del host.	Alta
SyRA2	Implementación estricta de zonas de seguridad diferenciadas: LAN Clientes (tráfico usuario), DMZ (Servidor Autoritativo) y TRANSIT (Servidor Recursivo).	Alta
SyRA3	Inclusión de dos routers Cisco C7200 (R1-BORDE e ISP-CORE) con enrutamiento estático y en un enlace punto a punto (10.0.0.0/30) que introduce latencia inter-zonas real (20ms por salto).	Alta
SyRA4	El nodo atacante (contenedor Docker con Python 3.11 y Scapy 2.7.0) debe ubicarse en la red de acceso LAN (192.168.10.0/24) conectado al mismo switch que los clientes, con visibilidad directa al servidor recursivo para ejecutar ataques de Flooding y Spoofing .	Alta

La disposición de la arquitectura híbrida (Docker + GNS3) responde a la necesidad de simular un escenario de DNS. Esta configuración permite evaluar cuantitativamente si la validación criptográfica DNSSEC (RRSIG verificado contra trust anchors) logra descartar las respuestas falsificadas, conforme a las recomendaciones del NIST SP 800-81r3 que estable la importancia de la validación de integridad en las respuestas DNS (NIST, 2026).

3.3 Arquitectura del Entorno Experimental

3.3.1. *Selección del Software de Simulación*

Este apartado se contextualiza en torno a la herramienta de simulación y emulación de redes, cuya selección se fundamenta en criterios de fidelidad técnica, capacidad de virtualización y soporte para la implementación de servicios de infraestructura crítica de Internet. La elección del software resulta determinante para garantizar la correcta ejecución de los escenarios de ataque y la validación de la cadena de confianza en DNSSEC.

La selección del entorno de laboratorio se establece como un elemento clave para la realización de los objetivos previstos para la implementación, permitiendo recrear de forma controlada una topología jerárquica que se asemeje a un DNS global (Root, TLD, Autoritativo) que opere con el mismo software y protocolos que los servidores reales. De esta manera, se lleva a cabo un análisis comparativo entre las principales herramientas disponibles en el ámbito académico y profesional, considerando los aspectos técnicos descritos en la sección anterior.

En la Tabla 7 se expone una división fundamental en las herramientas de evaluación de redes: la diferencia entre simuladores de eventos discretos y emuladores basados en la virtualización/orquestación de sistemas operativos reales.

Los simuladores NS-3 y OMNeT++ obtuvieron las puntuaciones más bajas (3/6) debido a su naturaleza algorítmica. Aunque poseen una alta escalabilidad teórica, modelan los protocolos mediante abstracciones matemáticas y código propio. Esto imposibilita el cumplimiento del requerimiento SRS4, ya que no permiten ejecutar binarios reales y de producción del demonio BIND9, sino aproximaciones simulación-dependientes. Asimismo, la integración nativa de herramientas de seguridad ofensiva basadas en la construcción de sockets crudos a bajo nivel con Python/Scapy (SRS5) requeriría el desarrollo de capas complejas de abstracción.

En el espectro de la emulación de sistemas reales, EVE-NG Community (5/6) representa una alternativa sólida, pero presenta una limitación crítica en el cumplimiento del criterio SRS1. Al requerir de forma generalizada una máquina virtual intermedia basada en Ubuntu para su despliegue sobre el sistema operativo anfitrión (EndeavourOS Linux), introduce una capa innecesaria de virtualización anidada (nested virtualization); lo cual genera un consumo redundante de recursos (overhead) en la CPU y la memoria RAM del host de simulación.

GNS3 se seleccionó como la plataforma definitiva al alcanzar la puntuación de 6/6. A diferencia de las otras alternativas, GNS3 se ejecuta de manera completamente nativa sobre el host Linux, permitiendo la interacción directa con el subsistema de red del núcleo (*kernel*). Esta arquitectura facilita la orquestación nativa de contenedores Docker (SRS2), los cuales comparten el kernel del sistema operativo anfitrión como se muestra en la Tabla 7.

Tabla 7

Herramientas de Evaluación de Redes.

Software de Simulación	**SRS1 (Host Linux Nativo)**	**SRS2 (Soporte Docker Nativo)**	**SRS3 (Escalabilidad n>40)**	**SRS4 (BIND 9 Real)**	**SRS5 (Herramienta Ataque Kali)**	**SRS6 (Análisis Tshark)**	**Total**
NS-3	1	0	1	0	0	1	3
OMNeT++	1	0	1	0	0	1	3
EVE-NG	0	1	1	1	1	1	5
GNS3	1	1	1	1	1	1	6

Nota: 1 = Cumple, 0 = No cumple

3.3.2. *Diseño de la Topología de Red.*

El diseño de la topología se fundamenta en el principio de segmentación de red y defensa en profundidad, según el cual el aislamiento de los servicios críticos en zonas diferenciadas reduce la superficie de ataque y limita el alcance de un eventual compromiso (NIST, 2026). En coherencia con este principio, la red se estructura en tres zonas separadas por enrutamiento, una zona de usuarios (LAN), una zona de tránsito y una zona desmilitarizada (DMZ) en la que residen los servidores. La decisión consiste en desacoplar el servidor DNS recursivo de la red de usuarios y ubicarlo en una zona distinta a la de los servidores autoritativos, en aplicación directa de la recomendación del NIST SP 800-81r3 de separar las funciones autoritativa y recursiva para reducir la superficie de ataque (NIST, 2026). Esta separación obliga, además, a que el tráfico DNS atraviese el dispositivo de enrutamiento, lo que permite medir el impacto de la latencia de red sobre la ventana de oportunidad del ataque de envenenamiento.

La topología incorpora también un esquema de alta disponibilidad mediante un servidor DNS autoritativo (maestro) y un servidor DNS esclavo. El autoritativo constituye la fuente de verdad de los registros del dominio y representa la jerarquía DNS superior, mientras que la réplica del esclavo, mantenida a través de la transferencia de zona (AXFR), permite evaluar si el ataque corrompe únicamente al servidor recursivo.

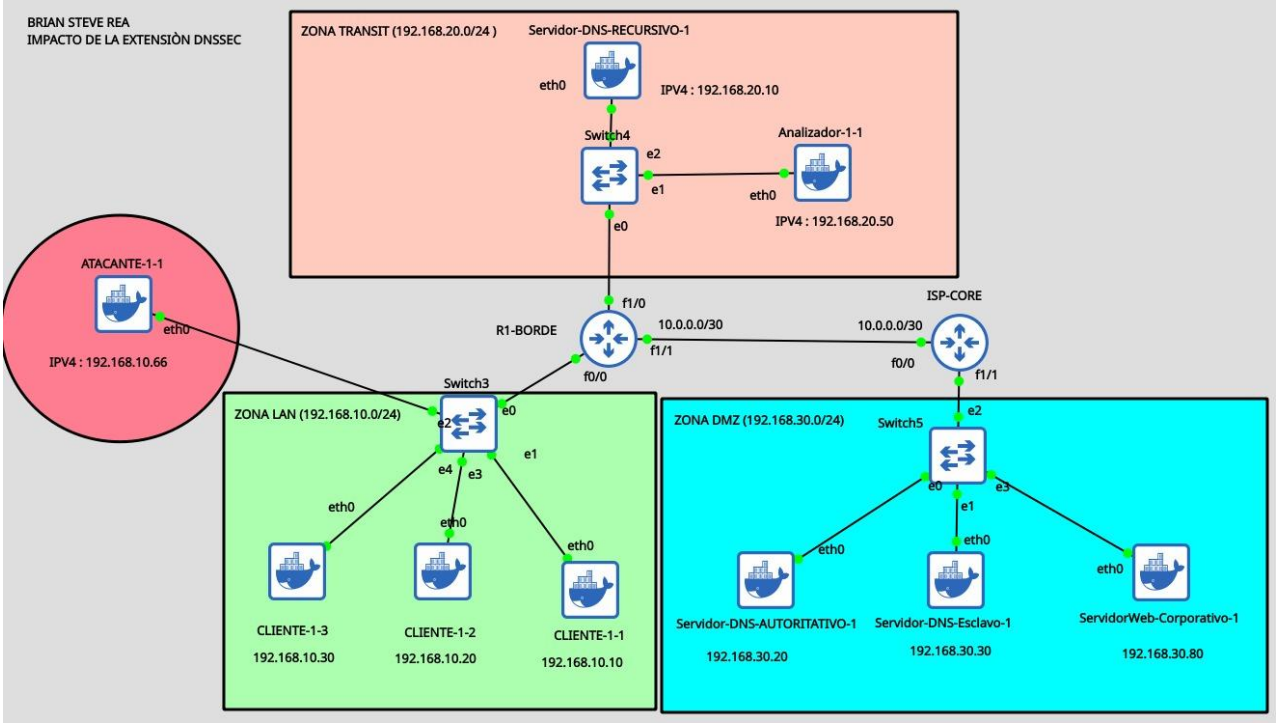
El modelo de amenaza adoptado posiciona al atacante dentro del segmento LAN, conectado al mismo switch que los usuarios legítimos, reproduciendo un escenario de amenaza interna (insider threat) o de host comprometido. Desde esa ubicación, el adversario debe atacar de forma remota al servidor recursivo a través del enrutador, intentando ganar la condición de carrera frente a las respuestas legítimas provenientes de los servidores autoritativo y esclavo, mecanismo característico del ataque de envenenamiento de caché descrito originalmente por Kaminsky

(2008). Dado que la reproducción de dicha condición de carrera resulta inviable en un entorno de laboratorio por la proximidad de los nodos y por la aleatorización de puerto de BIND 9.20, este trabajo adopta un modelo de inyección determinista, descrito en la sección de escenarios.

La topología de red implementada en GNS3 consta de tres zonas segmentadas mediante routers Cisco C7200, simulando una arquitectura corporativa con separación entre la red de usuarios (LAN), la red de tránsito y la zona desmilitarizada (DMZ) donde residen los servidores. El host EndeavourOS (20 núcleos de CPU y 15 GB de RAM asignados a GNS3) ejecuta la totalidad de los contenedores Docker. Los routers R1-BORDE e ISP-CORE proveen el enrutamiento inter-zona y la segmentación lógica que reproduce el aislamiento de red de un entorno corporativo. Cada nodo se implementa como un contenedor Docker basado en Debian (Bookworm), lo que habilita la persistencia de las configuraciones mediante volúmenes Docker y garantiza la reproducibilidad del entorno completo. Esta arquitectura híbrida en la que los contenedores se ejecutan como nodos de red dentro de GNS3 compartiendo el kernel del host elimina el sobrecosto de la virtualización completa y permite controlar la totalidad de las variables del experimento, en línea con el carácter reproducible exigido por el diseño experimental véase la Figura 8, que es la topología implementada en GNS3.

Figura 8

Arquitectura de red implementada en el simulador GNS3



Siguiendo con el diseño se muestra ahora los dispositivos que se encuentran dentro de la topología para identificar el rol que cumplen y también los sistemas operativos que son utilizados dentro de los Dockers seleccionados como se mira a continuación en la tabla 8.

Tabla 8

Dispositivos que conforman la topología de simulación.

Dispositivo	**RAM**	**CPU**	**Sistema Operativo**	**Rol en la Red**
DNS Recursivo	~250 MB	1	Debian 12 / BIND9 9.20.23 (Docker)	Resolutor recursivo, caché DNS y validación DNSSEC

DNS Autoritativo	~250 MB	1	Debian 12 / BIND9 9.20.23 (Docker)	Zona maestra empresa.local (90 registros)
DNS Esclavo	~200 MB	1	Debian 12 / BIND9 9.20.23 (Docker)	Réplica de zona, completar la arquitectura de alta disponibilidad.
Servidor Web	~50 MB	1	Nginx Alpine 1.26.3 (Docker)	Servicio HTTP corporativo
CLIENTE-1-1	~200 MB	1	Debian 12 (Docker)	Medición automatizada de latencia y throughput
CLIENTE-1-2	~200 MB	1	Debian 12 (Docker)	Tráfico concurrente de fondo
CLIENTE-1-3	~200 MB	1	Debian 12 (Docker)	Tráfico concurrente de fondo
ATACANTE-1-1	~200 MB	1	Python 3.11 / Scapy 2.7.0 (Docker)	Generador de carga adversaria y servidor DNS falso (dns_falso.py) para la demostración de respuesta forjada.
R1-BORDE	512 MB	1	Cisco IOS 15.2(4)S5 (Dynamips)	Enrutamiento LAN ↔ TRANSIT
ISP-CORE	512 MB	1	Cisco IOS 15.2(4)S5 (Dynamips)	Enrutamiento TRANSIT ↔ DMZ

Switch3 (LAN)	N/A	N/A	GNS3 Ethernet Switch	Conmutación zona LAN
Switch4 (TRANSIT)	N/A	N/A	GNS3 Ethernet Switch	Conmutación zona TRANSIT
Switch5 (DMZ)	N/A	N/A	GNS3 Ethernet Switch	Conmutación zona DMZ

3.4 Implementación del Entorno de simulación

3.4.1. *Especificaciones del Entorno.*

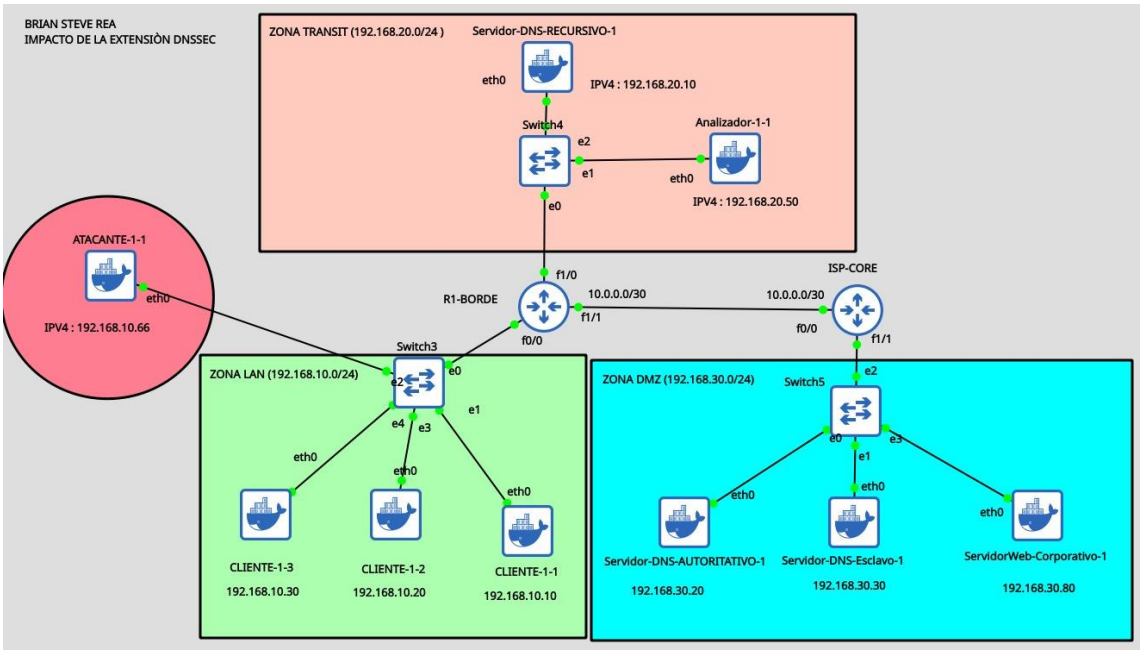
La implementación del entorno de simulación se realizó sobre la plataforma GNS3 versión 3.0.6, ejecutada en un equipo host con sistema operativo EndeavourOS (distribución basada en Arch Linux) con kernel 7.0.9. El equipo host cuenta con 20 núcleos de CPU y 15 GB de memoria RAM, recursos que permiten la ejecución simultánea de todos los nodos de la topología sin degradación de rendimiento.

Los nodos de la infraestructura DNS se implementaron como contenedores Docker integrados en GNS3, basados en la imagen python:3.11-slim con BIND9 9.20.23 preinstalado. Esta aproximación optimiza significativamente el consumo de recursos del host en comparación con máquinas virtuales completas, permitiendo asignar entre 150 y 250 MB de RAM por contenedor frente a los 1-2 GB que requeriría una máquina virtual con sistema operativo completo. El enrutamiento se realizó mediante routers Cisco C7200 simulados con Dynamips e IOS 15.2(4)S5 (imagen C7200-ADVIPSERVICESK9-M, RELEASE SOFTWARE fc1). Cada router se emuló sobre un chasis Cisco 7206VXR con procesador NPE-400 (CPU R7000 a 150 MHz),

512 MB de memoria DRAM, 32 MB de memoria compartida, 509 KB de NVRAM y 8 MB de Flash, con un registro de configuración 0x2102..

Figura 9

Topología de red implementada en GNS3 con segmentación por zonas.



Si bien la estructura física de la topología es representada en la Figura 9, se considera fundamental el desglose operativo de cada segmento del entorno. Con este propósito, en la Tabla 9 son detallados los componentes integrados y sus especificaciones de versión, facilitando así la selección de las herramientas de software compatibles para su gestión.

Tabla 9

Información de Componentes y su especificación.

Componente	Especificación
Plataforma de simulación	GNS3 3.0.6

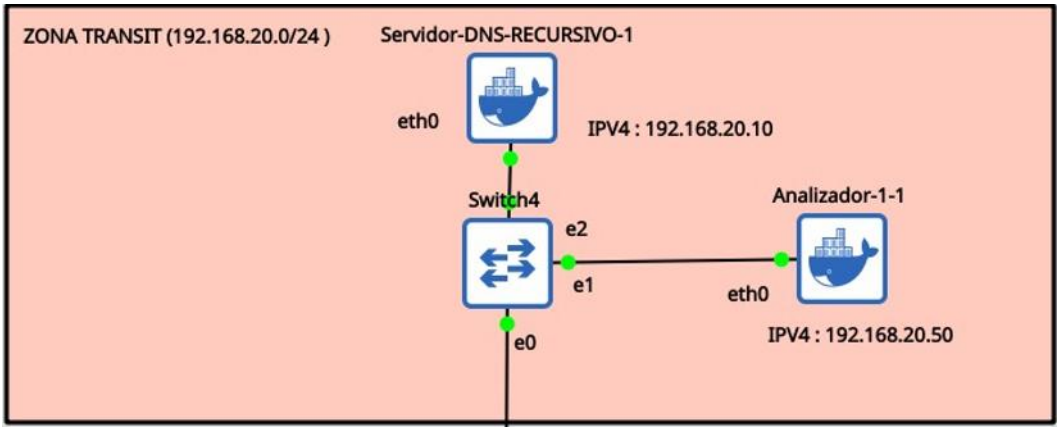
Motor de contenedores	Docker 29.5.1
Emulador de routers	Dynamips (Cisco C7200 IOS 15.2)
Sistema operativo host	EndeavourOS, kernel 7.0.9-arch2-1
Hardware host	20 CPU cores, 15 GB RAM
Servidor DNS	BIND9 9.20.23-1 (Debian 12)
Algoritmo DNSSEC	ECDSAP256SHA256 (algoritmo 13)
Contenedores Docker	9 (DNS, web, cliente, atacante, analizador)
Routers emulados	2 (R1-BORDE, ISP-CORE)
Switches virtuales	3 (LAN , TRANSIT, DMZ)

3.4.2. *Segmentación de red*

La topología se organizó en tres zonas interconectadas mediante dos routers Cisco C7200, replicando la segmentación típica de una red corporativa con DMZ. La zona de tránsito (192.168.20.0/24) alberga los servicios centrales de resolución: el servidor DNS recursivo (192.168.20.10), objetivo principal del ataque, y el analizador de tráfico (192.168.20.50) para la captura forense de paquetes..

Figura 10

Zona Transit donde se aloja el Servidor Recursivo y el Analizador de la simulación.



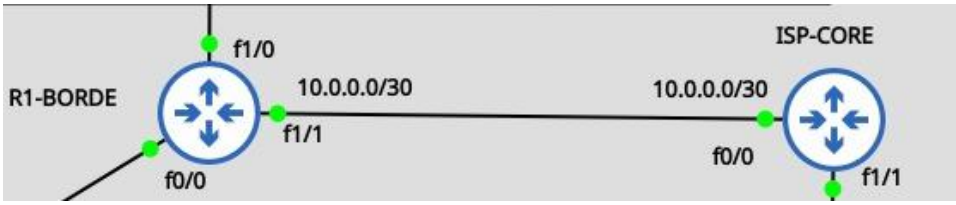
La Zona TRANSIT (192.168.20.0/24) fue creada como el segmento de observación de la resolución DNS, aislado deliberadamente del dominio de difusión donde reside el atacante con el fin de que las mediciones de rendimiento no se vieran contaminadas por tráfico ajeno al experimento. En ella se alojan los dos nodos centrales del análisis: el servidor DNS recursivo (192.168.20.10), objetivo principal del ataque y responsable de procesar las consultas, almacenarlas en caché y ejecutar la validación DNSSEC; y el nodo Analizador-1-1 (192.168.20.50), desde el cual se genera carga de consultas hacia el recursivo para evaluar su rendimiento sostenido. La medición es efectuada únicamente desde este nodo no captura la totalidad de las peticiones cursadas, las pruebas se complementan con la carga originada en los clientes de la zona LAN, de modo que el contraste entre ambos puntos de observación permite una caracterización más completa del comportamiento del servidor.

La interconexión interna de la zona se resuelve mediante el conmutador Switch4, elemento que mantiene a ambos nodos dentro del mismo segmento lógico sin proporcionarles conectividad directa hacia el resto de la topología.

Esta disposición responde a un criterio de segmentación ya que al carecer de conectividad directa con las zonas LAN y DMZ, todo el tráfico de consulta se ve obligado a atravesar el enrutador de borde antes de alcanzar al servidor recursivo, reproduciendo el recorrido que seguiría una petición en una red corporativa segmentada y preservando, un punto de captura limpio sobre la interfaz del recursivo.

Figura 11

Zona Backbone



El backbone de enrutamiento es el núcleo de interconexión entre los segmentos de red definidos en la topología. Su función es encaminar el tráfico entre las zonas LAN, TRANSIT y DMZ, las cuales se encuentran lógicamente separadas y no poseen conectividad directa entre sí.

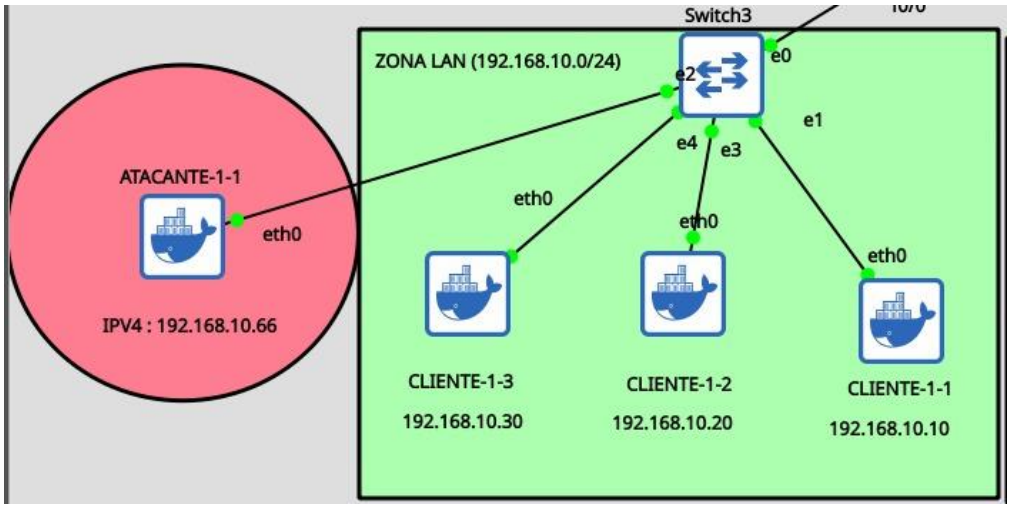
Para tal fin, fue desplegado el enrutador R1-BORDE, sobre el cual se concentran las interfaces que dan acceso a cada zona, la interfaz f0/0 hacia la zona LAN, la interfaz f1/0 hacia la zona TRANSIT y la interfaz f1/1 hacia el enrutador de frontera ISP-CORE.

La comunicación entre R1-BORDE e ISP-CORE se estableció mediante un enlace punto a punto sobre el segmento 10.0.0.0/30. Este segmento fue seleccionado por corresponder a la máscara de menor tamaño útil para un enlace de dos extremos, lo que permite el direccionamiento de ambos nodos sin desperdicio de espacio de red. A través de dicho enlace, el enrutador ISP-

CORE proporciona el acceso hacia la zona DMZ, donde residen los servidores autoritativo, esclavo y web corporativo.

Figura 12

Zona LAN Donde se representa los usuarios corporativos, el tráfico legítimo y la amenaza interna.



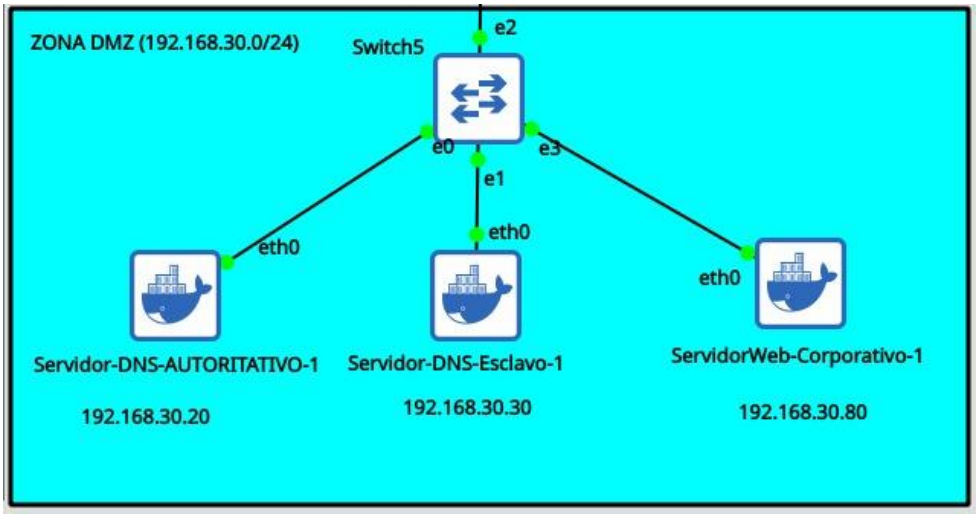
La Zona LAN (192.168.10.0/24) es la que representa la red de usuarios corporativos donde coexisten el tráfico legítimo y la amenaza interna. Se implementaron tres clientes DNS con roles diferenciados: CLIENTE-1-1 (192.168.10.10), un contenedor Docker que ejecuta las consultas de medición automatizada y genera datos CSV para el análisis estadístico; CLIENTE-1-2 (192.168.10.20), un contenedor Docker que genera tráfico DNS concurrente de fondo simulando múltiples usuarios corporativos; y CLIENTE-1-3 (192.168.10.30). El nodo atacante

(192.168.10.66) ejecuta las herramientas de ataque Scapy. La zona se conecta al router R1-BORDE mediante la interfaz f0/0 con gateway 192.168.10.1.

La Zona DMZ (192.168.30.0/24) es la que hospeda la infraestructura DNS autoritativa para el dominio empresa.local. Se implementó un esquema maestro-esclavo con un servidor DNS autoritativo primario (192.168.30.20) y un servidor esclavo (192.168.30.30) para evaluar la propagación de datos entre servidores. El servidor web corporativo (192.168.30.80) ejecuta Nginx y representa el recurso al que acceden los usuarios mediante resolución DNS. La zona DMZ se conecta al router ISP-CORE a través de la interfaz FastEthernet1/1 con gateway 192.168.30.1.

Figura 13

Componentes que se encuentran en la Zona DMZ (servidores).



La segmentación en zonas de seguridad diferenciadas permite evaluar los ataques en un escenario realista en el que el atacante comparte el dominio de difusión con las víctimas, a la vez

que se aísla el tráfico de los servidores. La Tabla 10 detalla la composición de cada zona, su segmento de red y su función dentro del entorno experimental.

Tabla 10

Composición de cada zona con su segmento de red y la función que cumple.

Zona	**Segmento IPv4**	**Nodos**	**Función**
ZONA LAN	192.168.10.0/24	CLIENTE-1-1 (.10), CLIENTE-1-2 (.20), CLIENTE-1-3 (.30), ATACANTE-1-1 (.66)	Red de usuarios y punto de origen del ataque
ZONA DE TRÁNSITO	192.168.20.0/24	Servidor DNS Recursivo (.10), Analizador (.50)	Resolución, validación DNSSEC y captura de tráfico
ZONA DMZ	192.168.30.0/24	DNS Autoritativo (.20), DNS Esclavo (.30), Servidor Web (.80)	Servidores de la zona empresa.local

Como parte de la configuración del entorno de simulación, se establece un esquema de direccionamiento IPv4 estructurado. La Tabla 11 detalla la asignación lógica y física de las direcciones IP para cada uno de los nodos y las interfaces de los enrutadores que conforman la topología. Este esquema se encuentra segmentado en cuatro zonas operativas principales (LAN,

TRANSIT, Enlace y DMZ), lo cual permite aislar los servicios y representar fielmente la arquitectura de una red corporativa conectada a un proveedor de servicios (ISP).

Tabla 11

Direccionamiento IP completo de la topología.

Nodo	**Dirección IP**	**Zona**	**Función**
R1-BORDE Fa0/0	192.168.10.1	LAN	Gateway LAN
R1-BORDE Fa1/0	192.168.20.1	TRANSIT	Gateway TRANSIT
R1-BORDE Fa1/1	10.0.0.1	Enlace	Punto a punto ISP
ISP-CORE Fa0/0	10.0.0.2	Enlace	Punto a punto R1
ISP-CORE Fa1/1	192.168.30.1	DMZ	Gateway DMZ
DNS Recursivo	192.168.20.10	TRANSIT	Resolución y caché
Analizador	192.168.20.50	TRANSIT	Pruebas de Throughput
CLIENTE-1-1	192.168.10.10	LAN	Medición de latencia y throughput
CLIENTE-1-2	192.168.10.20	LAN	Tráfico concurrente de fondo
CLIENTE-1-3	192.168.10.30	LAN	Tráfico de Fondo
Atacante	192.168.10.66	LAN	Suplantación de autoritativo y flooding
DNS Autoritativo	192.168.30.20	DMZ	Zona empresa.local
DNS Esclavo	192.168.30.30	DMZ	Réplica de zona
Servidor Web	192.168.30.80	DMZ	HTTP corporativo

De acuerdo con los parámetros expuestos en la Tabla 11, la distribución de los nodos facilita la evaluación del comportamiento del sistema DNS. En la zona de red local (LAN) son ubicados los clientes legítimos junto con el nodo del atacante, desde el cual serán orquestadas las peticiones maliciosas para forzar el envenenamiento de caché. Por otro lado, la zona TRANSIT alberga de forma aislada al servidor DNS Recursivo, encargado de procesar y almacenar las consultas.

En la zona desmilitarizada (DMZ) son alojados los recursos críticos, los servidores DNS Autoritativo y Esclavo responsables de la gestión de la zona *empresa.local*, así como el servidor web corporativo; garantizando que el tráfico de consulta pase obligatoriamente por el núcleo del ISP.

3.4.3. *Justificación de la Topología de Red*

La topología propuesta recrea un entorno operativo funcional, diseñado para validar la seguridad del protocolo DNS bajo condiciones controladas. La red se estructura mediante un enrutamiento central que separa la zona de usuarios (LAN), donde coexisten el tráfico legítimo y la amenaza, de la zona de servicios, en la que residen los servidores DNS y el servidor web. Esta separación entre las funciones autoritativas y recursivas se ajusta a la directriz del NIST SP 800-81r3, que recomienda aislar los servicios DNS para reducir la superficie de ataque (NIST, 2026).

Primero, se implementa un esquema de alta disponibilidad mediante la inclusión de un servidor DNS autoritativo (maestro) y un servidor DNS esclavo, el autoritativo constituye la fuente de verdad para los registros del dominio y representa la jerarquía DNS superior.

En segundo, se desacopla el servidor recursivo de la red de usuarios y se ubica en la zona de servicios, de modo que el tráfico DNS se ve obligado a atravesar el dispositivo de enrutamiento; con ello se replica un escenario empresarial o de ISP, en el que la infraestructura crítica reside en

un segmento distinto al de los clientes, y se habilita la medición del impacto de la latencia de red sobre la ventana de oportunidad del ataque.

Se incorpora un servidor web en la zona de servicios con el fin de trascender las métricas establecidas y cuantificar el éxito del ataque desde la perspectiva del usuario final, ya que una carga adversaria redirige a los clientes hacia un sitio fraudulento o les niega el acceso al servicio, materializando el riesgo en una suplantación o denegación verificable. El nodo atacante, por su parte, se posiciona en el segmento LAN, conectado al mismo switch que los usuarios legítimos, lo que reproduce un escenario de amenaza interna (*insider threat*) o de host comprometido, desde esa ubicación, el adversario ataca de forma remota al servidor recursivo a través del enrutador. La arquitectura centrada en el enrutador confiere, además, escalabilidad al diseño, al facilitar la futura incorporación de firewalls o sistemas de detección de intrusos (IDS) entre el switch de la LAN y el enrutador sin necesidad de reestructurar la topología base.

La cantidad de nodos cliente se define con el propósito de evaluar el impacto del envenenamiento en un entorno multiusuario, conforme a la recomendación del NIST SP 800-81r3 de analizar la resiliencia DNS bajo condiciones de concurrencia (NIST, 2026). Para ello se despliegan tres clientes en la zona LAN, con funciones diferenciadas. El primero, CLIENTE-1-1 (192.168.10.10, contenedor Docker), actúa como cliente de medición automatizada, este ejecuta los scripts de medición de latencia y througput, exportando los datos cuantitativos en formato CSV para el análisis estadístico. El segundo, CLIENTE-1-2 (192.168.10.20, contenedor Docker), funciona como generador de tráfico concurrente, produciendo consultas DNS continuas en segundo plano que simulan el patrón de múltiples usuarios corporativos y permiten verificar que el envenenamiento de la caché del recursivo afecta a todos los clientes que dependen de ese resolutor, con independencia de cuál haya originado la consulta interceptada. El tercero,

CLIENTE1-3 (192.168.10.30 contenedor Docker), también genera tráfico concurrente por lo que la concurrencia generada por estos tres nodos permite confirmar que el compromiso del resolutor se propaga simultáneamente a todos los usuarios de la red.

La implementación de la topología se realiza de manera híbrida sobre el emulador GNS3 versión 3.0.6, utilizando EndeavourOS (distribución basada en Arch Linux) como sistema operativo anfitrión. La pieza clave de la arquitectura es la integración nativa del demonio Docker 29.5.1 con GNS3, que permite ejecutar los contenedores como nodos de red dentro de la topología compartiendo el kernel del host y eliminando el sobrecosto de la virtualización completa. Se definen tres segmentos lógicos; en la zona LAN (192.168.10.0/24) se despliegan contenedores basados en la imagen python:3.11-slim de bajo consumo de recursos (aproximadamente 200 MB de RAM por contenedor), que alojan a los tres clientes y al atacante con las herramientas de red necesarias (dig, ping, curl) y la biblioteca Scapy 2.7.0 en el nodo ofensivo. Los servidores DNS se implementan mediante imágenes de BIND9 con las extensiones de seguridad activas, y la cadena de confianza se valida en el resolutor recursivo contra los anclajes de confianza (*trust anchors*) configurados, de modo que se rechace toda respuesta cuya firma RRSIG no se corresponda con el registro DS del nivel superior. Los dos routers Cisco C7200 (R1-BORDE e ISP-CORE) aportan una latencia inherente de aproximadamente 20 ms por salto debido al procesamiento de paquetes en Dynamips.

3.5 Implementación de los Nodos.

3.5.1. *Configuración del enrutamiento*

La interconexión entre las tres zonas de red se realizó mediante dos routers Cisco C7200 emulados con Dynamips, configurados con enrutamiento estático. El router R1-BORDE actúa como punto central de la topología con tres interfaces activas: FastEthernet0/0 (192.168.10.1)

como gateway de la zona LAN, FastEthernet1/0 (192.168.20.1) como gateway de la zona TRANSIT, y FastEthernet1/1 (10.0.0.1) como enlace punto a punto hacia ISP-CORE.

El router ISP-CORE proporciona conectividad hacia la zona DMZ con la interfaz FastEthernet1/1 (192.168.30.1) como gateway, y FastEthernet0/0 (10.0.0.2) como enlace hacia R1-BORDE. Se configuraron rutas estáticas bidireccionales: R1-BORDE tiene una ruta hacia 192.168.30.0/24 vía 10.0.0.2, e ISP-CORE tiene rutas hacia 192.168.10.0/24 y 192.168.20.0/24 vía 10.0.0.1.

Figura 14

Tabla de enrutamiento del router R1-BORDE

```
R1-BORDE#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.0.0.0/30 is directly connected, FastEthernet1/1
L        10.0.0.1/32 is directly connected, FastEthernet1/1
      192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.10.0/24 is directly connected, FastEthernet0/0
L        192.168.10.1/32 is directly connected, FastEthernet0/0
      192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.20.0/24 is directly connected, FastEthernet1/0
L        192.168.20.1/32 is directly connected, FastEthernet1/0
S     192.168.30.0/24 [1/0] via 10.0.0.2
```

En la figura se observan las redes directamente conectadas (C/L) de las zonas LAN (192.168.10.0/24), TRANSIT (192.168.20.0/24) y el enlace punto a punto (10.0.0.0/30), junto con la ruta estática (S) hacia la DMZ (192.168.30.0/24) vía 10.0.0.2.

Figura 15

Tabla de enrutamiento del router R1-CORE

```
ISP-CORE#sho ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.0.0.0/30 is directly connected, FastEthernet0/0
L        10.0.0.2/32 is directly connected, FastEthernet0/0
S     192.168.0.0/16 [1/0] via 10.0.0.1
S     192.168.10.0/24 [1/0] via 10.0.0.1
S     192.168.20.0/24 [1/0] via 10.0.0.1
      192.168.30.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.30.0/24 is directly connected, FastEthernet1/1
L        192.168.30.1/32 is directly connected, FastEthernet1/1
```

En la figura se muestra la red DMZ (192.168.30.0/24) directamente conectada y las rutas estáticas (S) hacia los segmentos LAN y TRANSIT vía 10.0.0.1.

Figura 16

Estado de las interfaces del router R1-BORDE.

```
R1-BORDE#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.10.1    YES NVRAM  up                    up
FastEthernet1/0        192.168.20.1    YES NVRAM  up                    up
FastEthernet1/1        10.0.0.1        YES NVRAM  up                    up
FastEthernet2/0        unassigned      YES NVRAM  administratively down down
FastEthernet2/1        unassigned      YES NVRAM  administratively down down
GigabitEthernet3/0     unassigned      YES NVRAM  administratively down down
```

En la figura 16 se muestran las tres interfaces activas (Fa0/0, Fa1/0, Fa1/1) presentan estado up/up con las direcciones de gateway asignadas a cada zona con el comando show ip interface brief.

Figura 17

Estado de las interfaces del router ISP-CORE

```
ISP-CORE#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        10.0.0.2        YES NVRAM  up                    up
FastEthernet1/0        unassigned      YES NVRAM  administratively down down
FastEthernet1/1        192.168.30.1    YES NVRAM  up                    up
FastEthernet2/0        unassigned      YES NVRAM  administratively down down
FastEthernet2/1        unassigned      YES NVRAM  administratively down down
GigabitEthernet3/0     unassigned      YES NVRAM  administratively down down
```

Con el comando show ip interface brief que se muestra en la Figura 17 las interfaces Fa0/0 (enlace punto a punto) y Fa1/1 (gateway DMZ) se encuentran up/up; las restantes permanecen administrativamente desactivadas.

La verificación de conectividad se realizó mediante pruebas de ping desde el cliente hacia todos los segmentos de red, confirmando tiempos de respuesta de 13-17 ms entre zonas LAN y TRANSIT (un salto de enrutamiento), y 20-30 ms entre la zona LAN y la DMZ (dos saltos de enrutamiento a través de R1-BORDE e ISP-CORE). Estos valores de latencia son consistentes con el procesamiento de paquetes en Dynamips y proporcionan datos para las mediciones de rendimiento DNS.

Tabla 12

Tabla de verificación de conectividad.

ORIGEN	DESTINO	SALTOS	LATENCIA	ESTADO
CLIENTE-1-1	REC (.20.10)	1	13-17 ms	OK
CLIENTE-1-1	AUT (.30.20)	2	20-30 ms	OK
REC (.20.10)	ATACANTE (.10.66)	1	~12–23 ms	OK

3.5.2. *Configuración del servidor DNS autoritativo.*

El servidor DNS autoritativo mantiene la autoridad sobre la zona empresa.local, que reproduce los servicios de una organización corporativa, y responde de forma no recursiva únicamente a las consultas correspondientes a los dominios de los que es autoritativo. La resolución recursiva se delega en un servidor independiente (dns-recursivo), lo que respeta la separación de roles recomendada en despliegues DNS reales.

La zona empresa.local se declaró como zona maestra (master) en el archivo de configuración de BIND. La transferencia de zona se restringió exclusivamente al servidor esclavo, evitando que un tercero pueda descargar la zona completa (mitigación de fuga de información por AXFR).

Bloque de Código 1

Declaración de la zona empresa.local en named.conf.local.

```
       zone "empresa.local" {
    type master;
    file "/etc/bind/db.empresa.local";
    allow-transfer { 192.168.30.30; };   // solo el DNS esclavo
    also-notify   { 192.168.30.30; };    // notificación de cambios
};
```

En la sección de opciones globales (named.conf.options) se deshabilitó la recursión, garantizando que el servidor actúe únicamente como autoritativo:

Bloque de Código 2

Opciones relevantes en named.conf.options.

```
       options {
    directory "/var/cache/bind";
    recursion no;              // servidor autoritativo puro
```

```
    allow-query { any; };
    dnssec-validation no;      // no valida: no es resolutor
    listen-on { any; };
};
```

El archivo de zona (db.empresa.local) contiene el registro SOA, los registros NS que declaran los servidores autoritativos, y los registros de dirección (A) y alias (CNAME) de cada servicio. El valor TTL se fijó en 300 segundos para todos los registros: un valor bajo que favorece la observación del comportamiento del caché (escenarios de caché fría y caliente) sin retener respuestas obsoletas durante las pruebas. En un entorno de producción, los TTL habituales oscilan entre 3 600 y 86 400 segundos.

Bloque de Código 3

Extracto del archivo de zona db.empresa.local.

```
       $TTL 300
@   IN  SOA dns-autoritativo.empresa.local. admin.empresa.local. (
            2024060801   ; serial   (AAAAMMDD + contador)
            3600         ; refresh  (consulta del esclavo)
            1800         ; retry    (reintento si falla)
            604800       ; expire   (descarte de zona)
            300 )        ; minimum  (TTL negativo)

@               IN  NS    dns-autoritativo.empresa.local.
@               IN  NS    dns-esclavo.empresa.local.
web-corporativo IN  A     192.168.30.80
www             IN  CNAME web-corporativo
correo          IN  A     192.168.30.81
erp             IN  A     192.168.30.90
google          IN  A     192.168.30.100
; ... (registros A restantes)
```

Como se muestra en el Bloque de Código 3 está los 300 segundos de TTL y los dominios implementados dentro del Servidor Autoritativo que van a ser ocupados para las pruebas.

Los campos temporales del registro SOA gobiernan la relación maestro-esclavo, para lo cual debemos conocer la interpretación que estos tienes y se muestran dentro de la Tabla 13 que está a continuación.

Tabla 13

Interpretación de los campos temporales del registro SOA.

Campo	Valor	Función
serial	2024060801	Versión de la zona; se incrementa en cada cambio para disparar la sincronización del esclavo.
refresh	3600	Intervalo (s) con que el esclavo consulta si hay cambios.
retry	1800	Tiempo (s) de reintento si el refresh falla.
expire	604800	Tiempo (s) tras el cual el esclavo descarta la zona si no contacta al maestro.
minimum	300	TTL para respuestas negativas (NXDOMAIN).

La zona empresa.local agrupa los servicios simulados por categoría funcional. La zona contiene un total de 90 registros de tipo A, distribuidos principalmente en el segmento de la DMZ (192.168.30.0/24), además de los registros correspondientes a la infraestructura DNS, los clientes de la LAN y los nodos de prueba. La Tabla 14 resume los dominios configurados agrupados por categoría y su rango de direcciones.

Tabla 14

Dominios configurados en la zona empresa.local (90 registros).

Categoría	Dominios representativos	Rango de IP

Infraestructura DNS	dns-autoritativo, dns-esclavo, dns-recursivo	192.168.30.20, .30; 192.168.20.10
Servicios web	web-corporativo, webmail, portal, landing, tienda	192.168.30.80 – .84
Comunicaciones	correo, mail, smtp, imap, chat, videoconf, pbx, mensajería	192.168.30.85 – .89
Colaboración	intranet, wiki, docs, cloud, sharepoint	192.168.30.90 – .94
Gestión empresarial	erp, crm, nómina, contabilidad, inventario, facturación, rrhh, compras, tesorería	192.168.30.95 – .105
Seguridad	vpn, firewall, proxy, ldap, radius, siem, ids, waf	192.168.30.106 – .113
DevOps e integración	docker-registry, ci-cd, ansible, terraform, nexus, sonarqube	192.168.30.114 – .119
Monitorización	grafana, prometheus, elasticsearch, kibana, zabbix, nagios, monitor	192.168.30.120 – .123, .160 – .162
Servicios de red	ntp, dhcp, tftp, syslog, snmp-manager	192.168.30.124 – .128
Almacenamiento	backup, storage, nas	192.168.30.129 – .131
Bases de datos	mysql, postgres, mongodb, redis, oracle, mariadb	192.168.30.132 – .137
Servicios de aplicación	api, api-gateway, cdn, caché, queue, microservicio1, microservicio2, load-balancer	192.168.30.138 – .145

Sitios externos simulados	google, facebook, banco, gobierno, universidad, sri, iess, cnt	192.168.30.150 – .157
Nodos de red (clientes y prueba)	cliente-1, cliente-2, cliente-3, atacante, analizador	192.168.10.10, .20, .30, .66; 192.168.20.50

Algunos nombres comparten dirección IP al representar alias funcionales del mismo servicio (por ejemplo, correo y mail; chat y mensajería), por lo que el número de nombres supera al de direcciones únicas. El conteo total de registros A de la zona se verifico de forma directa sobre el archivo de zona, con los comandos que se indica en el Bloque de Código 4.

Antes de poner el servicio en operación se validó la sintaxis de la zona y del archivo de configuración mediante las herramientas de comprobación integradas de BIND. La salida OK, junto con el número de registros reportado, constituye la evidencia de que la zona carga sin errores.

Bloque de Código 4

Comandos de validación de la zona y la configuración.

```
# Valida la sintaxis y cuenta los registros de la zona
named-checkzone empresa.local /etc/bind/db.empresa.local

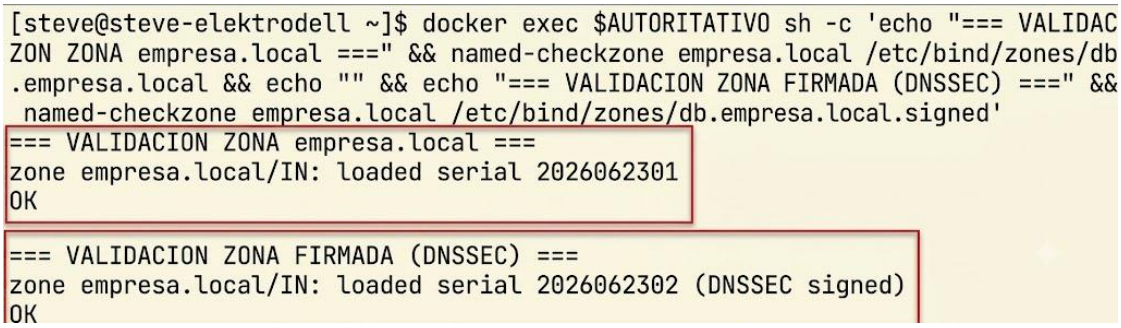
# Valida el archivo de configuración global
named-checkconf


# Cuenta los registros de tipo A de la zona
grep -E “\sIN\s+A\s” /etc/bind/db.empresa.local | wc -1
```

Para verificar la validez de las firmas, debemos observar que la validación de la zona y la zona firma se encuentren dentro del Servidor Autoritativo.

Figura 18

Validación de la zona empresa.local con named-checkzone.



La zona se replica hacia un servidor esclavo mediante transferencia AXFR. El maestro notifica los cambios (also-notify) y solo permite la transferencia al esclavo autorizado (allow-transfer). Se verificó que el esclavo recibió la zona completa consultando directamente su copia.

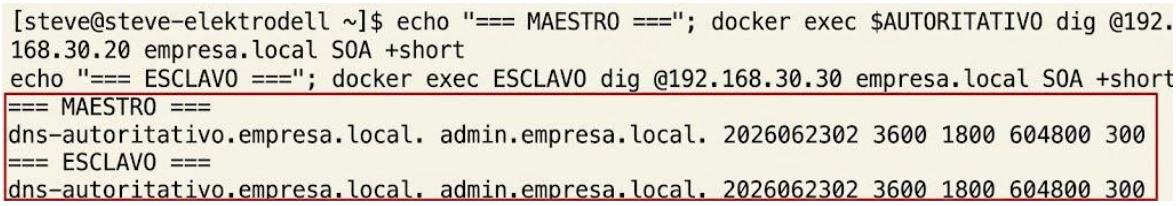
Bloque de Código 5

Verificación de la transferencia de zona hacia el esclavo.

```
dig AXFR empresa.local @192.168.30.30
```

Figura 19

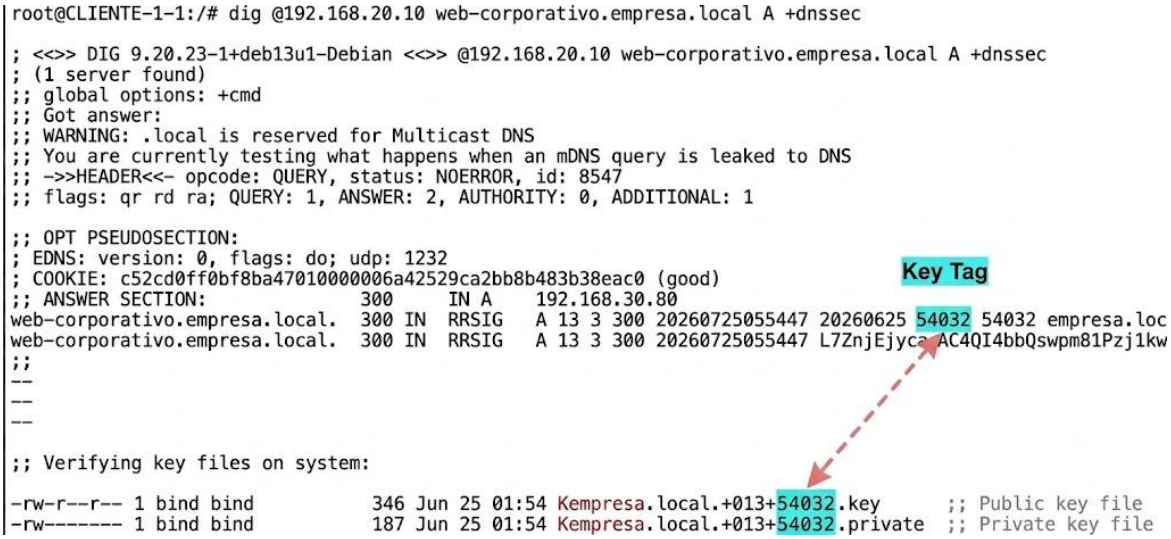
Transferencia de Zona recibida en el Servidor Esclavo.



Se comprueba la resolución de nombres desde el cliente (CLIENTE-1-1) hacia el servidor autoritativo, confirmando que las consultas de los dominios de empresa.local devuelven la dirección A esperada. Con los comandos que tenemos a continuación `dig @192.168.30.20 web-corporativo.empresa.local A`, que se puede ver en la Figura 20.

Figura 20

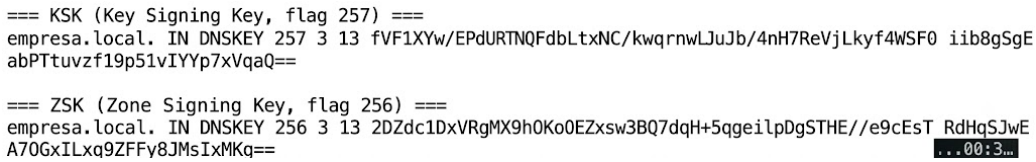
Verificación de resolución DNS correcta desde el cliente.



La variante firmada de esta misma zona se emplea en los escenarios E2 y E4. La zona se firmó con el algoritmo ECDSAP256SHA256, utilizando una clave de firma de clave (KSK, key tag 28475) y una clave de firma de zona (ZSK, key tag 54032). El resultado es el archivo de zona firmado (con extensión .signed) y el registro DS correspondiente. Al tratarse de un dominio de uso interno (.local), que no está delegado desde la raíz DNS, no existe una zona padre que publique el registro DS; por ello, el ancla de confianza se configura de forma manual en el resolutor validador en lugar de establecerse mediante una cadena DS descendente desde la raíz. El detalle de la generación de claves, el proceso de firma y la evidencia de validación (dig +dnssec) se documenta en la sección correspondiente al escenario E2 que se encuentra más adelante.

Figura 21

Zona empresa.local firmada.



En la Figura 21 están los registros DNSKEY de la zona empresa.local generados con el algoritmo ECDSAP256SHA256 (algoritmo 13). La Key Signing Key (KSK, identificador 28475) se distingue por el flag 257 y constituye el ancla de confianza de la zona, mientras que la Zone Signing Key (ZSK, identificador 54032) presenta el flag 256 y firma los registros de recursos. La separación de ambas claves permite rotar la ZSK sin alterar el ancla de confianza configurada en el resolutor.

Se muestra el resultado en la figura 22 sobre el proceso de firmado de la zona empresa.local. La validación mediante named-checkzone confirma que la zona firmada carga sin errores de sintaxis ni de estructura, incorporando los registros de seguridad DNSSEC (RRSIG, DNSKEY y NSEC3) sobre la totalidad de los registros de recursos de la zona.

Figura 22

Zona Firma con RRSIG.

```
[steve@steve-elektrodell ~]$ docker exec $AUTORITATIVO named-checkzone -D empresa.local /etc/bind/
ones/db.empresa.local.signed 2>/dev/null | head -25
empresa.local.                          300 IN SOA      dns-autoritativo.empresa.local. adm
in.empresa.local. 2026062302 3600 1800 604800 300
empresa.local.                          300 IN RRSIG    SOA 13 2 300 20260729101401 2026062
9101401 54032 empresa.local. 6tvy992Xg+JFFBXNnLA/274bhftcmpjjhwS3wip+gHxv1Gml1bZPKSxR QUpsIed4BR9c3
nSMvhjDJD8cBq+1qg==
; resign=20260729101401
empresa.local.                          300 IN NS       dns-esclavo.empresa.local.
empresa.local.                          300 IN NS       dns-autoritativo.empresa.local.
empresa.local.                          300 IN RRSIG    NS 13 2 300 20260729101401 20260629
101401 54032 empresa.local. 3CM8h9z8Hp9PZQ1toemwIX1tnD2YFWaxibDSHETYLq91ZQd/G6yF3pn +aus64tyJex877
V63+IQGND1YqtwVQ==
; resign=20260729101401
empresa.local.                          300 IN MX       10 mail.empresa.local.
empresa.local.                          300 IN MX       20 smtp.empresa.local.
empresa.local.                          300 IN RRSIG    MX 13 2 300 20260729101401 20260629
```

En la Figura 22 se muestra el detalle de la zona firmada en el que cada conjunto de registros de recursos aparece acompañado de su correspondiente firma RRSIG, generada con la Zone Signing Key. La presencia de estas firmas es la condición que permite al resolutor validador verificar la integridad y autenticidad de cada respuesta.

Figura 23

Conteo de registros DNSSEC.

```
[steve@steve-elektrodell ~]$ docker exec $AUTORITATIVO sh -c
resa.local /etc/bind/zones/db.empresa.local.signed 2>/dev/nu
amed-checkzone -D empresa.local /etc/bind/zones/db.empresa.l
Y)"; echo "NSEC3: $(named-checkzone -D empresa.local /etc/bi
/null | grep -c NSEC3)"'
RRSIG: 371
DNSKEY: 5
NSEC3: 186
```

La cuantificación de los registros de seguridad generados durante el firmado de la zona empresa.local es la siguiente: las firmas RRSIG, que autentican cada conjunto de recursos; los registros DNSKEY, que publican las claves públicas de la zona; y los registros NSEC3, que proporcionan la prueba de negación auntenticada de existencia, previniendo la enumeración de la zona. El número exacto de cada tipo de registro se obtiene directamente del archivo de zona firmado, tal como se muestra en la figura 23.

3.5.3. *Configuración DNS esclavo.*

El servidor DNS esclavo (Servidor-DNS-Esclavo-1, 192.168.30.30) se implementó en la zona DMZ como réplica del servidor autoritativo maestro, con el fin de dotar al dominio empresa.local de un esquema de alta disponibilidad. Su configuración define la zona en modo esclavo, estableciendo al autoritativo primario (192.168.30.20) como origen de la transferencia.

Bloque de código 6

Configuración del origen de la transferencia en el servidor Esclavo.

```
zone "empresa.local" {
    type slave;
    masters { 192.168.30.20; };
    file "/var/cache/bind/db.empresa.local";
};
    file "/var/cache/bind/db.empresa.local";
};
```

```
zone "30.168.192.in-addr.arpa" {
   type slave;
   masters { 192.168.30.20; };
   file "/var/cache/bind/db.192.168.30";
};
```

La transferencia de zona (AXFR) desde el maestro se verificó mediante la carga correcta de la zona en el esclavo, confirmada con rndc zonestatus (estado secondary, serial 2026072202, zona firmada con DNSSEC). La copia replicada se almacena en formato raw en el directorio de caché y su tamaño (aproximadamente 4,6 MB) corresponde a la zona ampliada a 10.090 registros empleada en las pruebas de rendimiento (véase la sección de throughput).

La disponibilidad de esta réplica firmada cumple una función adicional en el contexto de las pruebas de seguridad: al conservar una copia idéntica y validable de la zona legítima, actúa como referencia de integridad que permite contrastar que la respuesta forjada recibida por el cliente durante el escenario de envenenamiento no coincide con el dato autoritativo real de la zona.

3.5.4. *Configuración del servidor DNS recursivo*

El servidor DNS recursivo se implementó mediante un contenedor Docker que ejecuta BIND 9.20.23 sobre Debian 12, alojado en la zona TRANSIT (192.168.20.10). A diferencia del servidor autoritativo, este nodo sí realiza resolución recursiva: es el servidor al que apuntan los clientes de la red, y el encargado de resolutor, cachear y cuando corresponde validar criptográficamente las respuestas mediante DNSSEC. Por su posición en el flujo de resolución, constituye el punto de observación del comportamiento del caché y el objetivo del ataque de envenenamiento evaluado en la investigación.

En las opciones globales (named.conf.options) se habilitó la recursión, se restringieron las redes autorizadas a consultar y recursar, y se activó la validación DNSSEC. Esta configuración contrasta con la del servidor autoritativo, donde la recursión está deshabilitada, la separación de roles es una decisión de diseño alineada con las buenas prácticas de despliegues DNS reales.

Y ahora vemos un bloque de configuración dentro de la carpeta named.conf.options donde se encuentran las configuraciones realizadas en el contenedor del Servidor Recursivo.

Bloque de código 7

Configuración del recursivo named.conf.options.

```
        options {
            directory "/var/cache/bind";
            recursion yes;
            allow-query { 192.168.10.0/24; 192.168.20.0/24; 127.0.0.1; };
            allow-recursion { 192.168.10.0/24; 192.168.20.0/24; 127.0.0.1; };
            dnssec-validation yes;    // Activar para E2 y E4
            forwarders { 8.8.8.8; };
            forward only;
            max-cache-size 64m;
            max-cache-ttl 300;
        };
```

La recursión se permite únicamente desde las redes 192.168.10.0/24 y 192.168.20.0/24, replicando la práctica de seguridad de restringir el acceso recursivo a redes internas de confianza. El caché se limita a 64 MB con un TTL máximo de 300 segundos, lo que permite observar el efecto del almacenamiento temporal en las mediciones de latencia sin que la persistencia excesiva de registros interfiera con las pruebas de envenenamiento.

Dado que empresa.local no es un dominio de nivel superior real y no cuelga de la jerarquía DNS global, el servidor recursivo no puede localizar la zona por delegación. Por ello se configuró una zona de tipo forward que reenvía todas las consultas de empresa.local directamente al servidor autoritativo.

Bloque de código 8

Zona de reenvío hacia el servidor autoritativo.

```
zone "empresa.local" {
    type forward;
    forward only;
    forwarders { 192.168.30.20; };   // IP del DNS autoritativo
};
```

La validación DNSSEC requiere un ancla de confianza. Como empresa.local es una isla de confianza que no está delegada desde la raíz DNS real, no basta con habilitar la validación, es

necesario proporcionar explícitamente al recursivo la clave pública de firma de clave (KSK) de la zona. Sin este bloque, el recursivo no podría validar las firmas de empresa.local y, en consecuencia, no podría rechazar una respuesta falsa: es la pieza que sostiene la prueba funcional del escenario E4 y se muestra en el Bloque de Código a continuación como estaría la firma de la clave.

Bloque de código 9

Ancla de confianza estática para la zona empresa.local.

```
trust-anchors {
    empresa.local. static-key 257 3 13
        "AwEAA...";     // clave pública KSK (key tag 28475)
};
```

La siguiente tabla resume las directivas clave del servidor recursivo y contrasta su valor frente al servidor autoritativo, evidenciando la separación de roles.

Tabla 15

Directivas clave del servidor recursivo frente al autoritativo.

Directiva	Recursivo	Autoritativo	Función
recursion	yes	no	Habilita la resolución recursiva para los clientes.
dnssec-validation	yes	no	Valida criptográficamente las firmas DNSSEC.
trust-anchors	definido	n/a	Ancla de confianza de la isla empresa.local.
forwarders	definido	n/a	Reenvía empresa.local al autoritativo (objeto del ataque).
allow-recursion	redes LAN	n/a	Restringe qué redes pueden recursar.

El servidor recursivo almacena en caché las respuestas resueltas durante el tiempo indicado por el TTL (300 segundos). Esta característica es importante, la primera consulta de un dominio implica recursión completa y, en el caso de DNSSEC, validación criptográfica (estado de caché fría); las consultas siguientes se responden directamente desde memoria (estado de caché caliente). Para medir la latencia de caché fría de forma repetible entre cada medición, se vació el caché del resolutor antes de cada consulta inicial.

Bloque de código 10

Comandos de gestión del caché del resolutor.

```
# Vacia el cache -> fuerza estado de cache fria en la siguiente consulta
rndc flush

# (Opcional) Vuelca el contenido del cache para inspeccion
rndc dumpdb -cache
```

Se validó la configuración global y se comprobó tanto la resolución normal como la validación DNSSEC. La prueba visual de que el recursivo valida correctamente las firmas es la presencia del flag ad (authenticated data) en la respuesta a una consulta con la opción +dnssec.

Bloque de Código 11

Validación de la configuración y comprobación de resolución.

```
# Valida el archivo de configuracion global
named-checkconf
# Resolucion normal desde el cliente a traves del recursivo
dig web-corporativo.empresa.local @192.168.20.10 +short
# Comprobacion de validacion DNSSEC (buscar el flag "ad")
dig web-corporativo.empresa.local @192.168.20.10 +dnssec
```

3.5.5. *Servidor Web Corporativo.*

El servidor web corporativo (ServidorWeb-Corporativo-1, 192.168.30.80) representa el recurso final al que acceden los usuarios mediante la resolución del nombre web-

corporativo.empresa.local. Se implementó sobre nginx 1.26.3, sirviendo una página corporativa estática en el directorio `/var/www/html`. Su función en el experimento es materializar el impacto del ataque desde la perspectiva del usuario final: en condiciones normales, la resolución dirige al cliente hacia esta dirección legítima (192.168.30.80), mientras que bajo un ataque exitoso el cliente sería redirigido hacia la dirección del atacante (192.168.10.66), evidenciando la suplantación a nivel de aplicación.

Figura 24

Página web de la empresa.local en html.

```
[steve@steve-elektrodell ~]$ docker exec "$CLI1" sh -c 'wget -qO- http://192.168.30.80/ 2>/dev/null | head -15'
<!DOCTYPE html>
<html>
<head>
  <title>Empresa Corporativa — Servidor Legítimo</title>
  <style>
    body{font-family:sans-serif;text-align:center;padding:60px;background:#f0f4f8;color:#2d3748}
    .card{background:white;border-radius:12px;padding:40px;max-width:500px;margin:0 auto;box-shadow:0 4px 20px rgba(0,
0,0,.1)}
    h1{color:#2b6cb0;margin-bottom:10px}
    .badge{background:#c6f6d5;color:#276749;padding:6px 14px;border-radius:20px;font-size:14px;display:inline-block;ma
rgin:10px 0}
    p{color:#718096;font-size:15px}
  </style>
</head>
<body>
  <div class="card">
    <h1>  Empresa Corporativa</h1>
```

3.5.6. *Nodo Analizador de Tráfico.*

El nodo analizador (Analizador-1-1, 192.168.20.50) se ubicó en la zona TRANSIT, en el mismo segmento de red que el servidor DNS recursivo, con el propósito de realizar la captura del tráfico DNS sin introducir la latencia del enrutamiento entre zonas. Se equipó con las herramientas tcpdump y tshark para el registro y análisis de paquetes, y con dnsperf para la generación de carga controlada. Su ubicación resultó es determinante ya que permite medir el rendimiento sin el tránsito a través del enrutador de borde, hizo posible discriminar entre la capacidad de procesamiento del servicio DNS y la limitación impuesta por la infraestructura de red emulada.

3.5.7. *Implementación de DNSSEC*

La implementación de DNSSEC se realiza en dos etapas: firma de la zona en el servidor autoritativo y configuración de la validación en el servidor recursivo. El algoritmo seleccionado es ECDSAP256SHA256 (algoritmo 13), conforme a la recomendación del NIST SP 800-81r3 que favorece algoritmos de curva elíptica por su menor tamaño de firma (64 bytes frente a 256+ bytes en RSA-2048) y rendimiento superior en la validación.

Generación de claves y firma de la zona (servidor autoritativo). Se generan dos pares de claves criptográficas: una KSK (Key Signing Key, flag 257) que firma el conjunto de claves DNSKEY, y una ZSK (Zone Signing Key, flag 256) que firma los registros individuales de la zona.

Bloque de código 12

Generación de claves y firma de la zona.

```
      # Generar ZSK y KSK
      cd /etc/bind/keys
      dnssec-keygen -a ECDSAP256SHA256 -n ZONE empresa.local
      dnssec-keygen -a ECDSAP256SHA256 -f KSK -n ZONE empresa.local

      # Incluir claves en el archivo de zona
      echo     '$INCLUDE     /etc/bind/keys/Kempresa.local.+013+XXXXX.key'     >>
db.empresa.local

      # Firmar la zona con NSEC3 (RFC 5155)
      dnssec-signzone -A -3 $(openssl rand -hex 8) \
          -N INCREMENT -o empresa.local \
          -K /etc/bind/keys -t db.empresa.local
```

La firma genera el archivo db.empresa.local.signed que contiene los registros originales más los registros RRSIG (firmas digitales), DNSKEY (claves públicas), NSEC3 (denegación de existencia autenticada) y NSEC3PARAM. Se utiliza NSEC3 en lugar de NSEC para proteger contra la enumeración de zona (zone walking), conforme al RFC 5155. El salt aleatorio se genera con openssl rand -hex 8 dado que xxd no está disponible en el contenedor Debian 12.

Se ilustra la configuración de la validación en el servidor recursivo donde se configura la directiva dnssec-validation yes junto con un bloque trust-anchors que contiene la clave pública KSK con la palabra clave static-key.

Bloque de código 13

Trust anchor en el servidor recursivo.

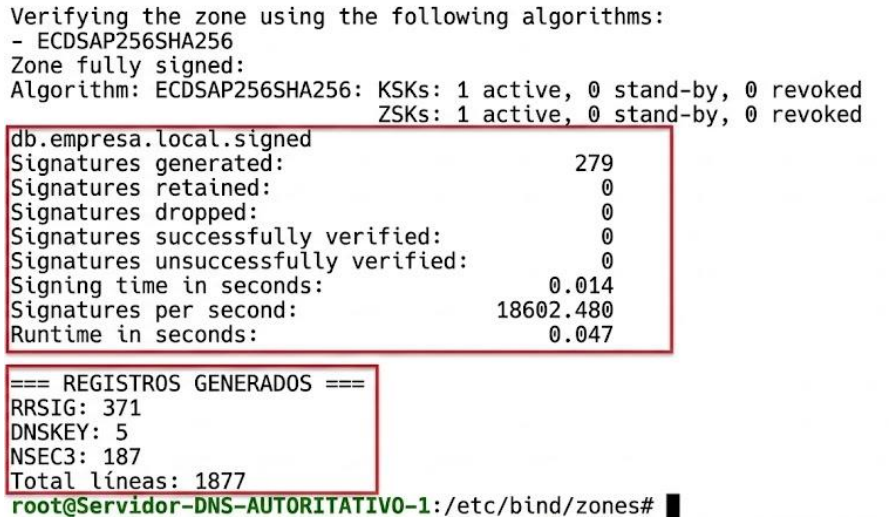
```
trust-anchors {
    "empresa.local." static-key 257 3 13
        "[clave pública KSK del autoritativo]";
};
```

La elección de static-key en lugar de initial-key es crítica para el funcionamiento en un entorno de laboratorio. La palabra clave initial-key implementa el mecanismo RFC 5011 (Automated Updates of DNSSEC Trust Anchors), que asume una cadena de confianza desde la zona raíz pública y busca registros DS en la zona padre. Dado que empresa.local es una zona privada sin delegación en la jerarquía DNS pública, initial-key genera errores de tipo "broken trust chain" al no encontrar registros DS en local. Con static-key, el recursivo confía directamente en la KSK configurada sin intentar validar la cadena ascendente.

La verificación de que DNSSEC funciona correctamente se realiza mediante el comando dig con el flag +dnssec. Una respuesta exitosa presenta tres indicadores: el flag AD (Authenticated Data) en la cabecera de respuesta, registros RRSIG en la sección de respuesta, y un tamaño de paquete mayor (~202 bytes frente a ~102 bytes sin DNSSEC).

Figura 25

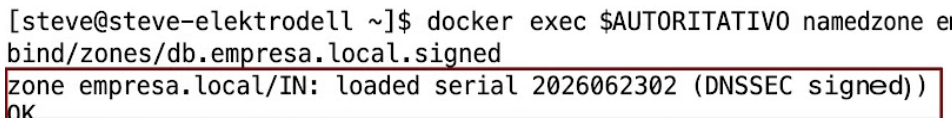
Firma exitosa de la zona empresa.local con DNSSEC



Una vez firmada la zona, se valida su integridad mediante named-checkzone sobre el archivo db.empresa.local.signed. La salida confirma que la zona firmada carga correctamente, incorporando los registros de seguridad sobre la totalidad de los registros de recursos.

Figura 26

Zona empresa.local firmada.



En la figura 26 se muestra la validación de la zona firmada mediante `named-checkzone`. La salida confirma que la zona carga sin errores de sintaxis ni estructura tras la incorporación de los registros DNSSEC.

El proceso de firmado generó los registros de seguridad DNSSEC sobre la zona. La zona firmada contiene, además de los 90 registros A originales, los registros RRSIG que autentican cada

conjunto de recursos, los registros DNSKEY con las claves públicas de la zona, y los registros NSEC3 y NSEC3PARAM que proporcionan la denegación de existencia autenticada.

Figura 27

Zona firmada con RRSIG.

```
[steve@steve-elektrodell ~]$ docker exec $AUTORITATIVO sh -c 'named-checkzone -D empresa.l
ocal /etc/bind/zones/db.empresa.local.signed 2>/dev/null | grep -A1 -E "web-corporativo|er
p|vpn" | head -20'
erp.empresa.local.                      300 IN A          192.168.30.97
erp.empresa.local.                      300 IN RRSIG      A 13 3 300 20260729101401
20260629101401 54032 empresa.local. uw0cnDkk7R70Hek+U9WJ7Uy98iPylKdwyzYBk2S+2/127v0tb8Kkms
rq /BGdmJz7rhrwyu7bGfDQoPt8j6l+DA==
; resign=20260729101401
```

En la figura 27 se muestra el detalle de la zona firmada en el que cada conjunto de registros de recursos aparece acompañado de su firma RRSIG, generada con la Zone Signing Key (identificador 54032).

Figura 28

Conteo de registros DNSSEC.

```
[steve@steve-elektrodell ~]$ docker exec $AUTORITATIVO sh -c 'echo "=== Registros DNSSEC e
n la zona firmada ==="; echo "RRSIG (firmas): $(named-checkzone -D empresa.local /etc/bind
/zones/db.empresa.local.signed 2>/dev/null | grep -c RRSIG)"; echo "DNSKEY (claves): $(nam
ed-checkzone -D empresa.local /etc/bind/zones/db.empresa.local.signed 2>/dev/null | grep -
c DNSKEY)"; echo "NSEC3 (negacion): $(named-checkzone -D empresa.local /etc/bind/zones/db.
empresa.local.signed 2>/dev/null | grep -c NSEC3)"; echo "NSEC3PARAM: $(named-checkzone -D
 empresa.local /etc/bind/zones/db.empresa.local.signed 2>/dev/null | grep -c NSEC3PARAM)"'
=== Registros DNSSEC en la zona firmada ===
RRSIG (firmas): 371
DNSKEY (claves): 5
NSEC3 (negacion): 186
NSEC3PARAM: 3
```

En la figura 28 se da a conocer los registros de seguridad generados durante el firmado: firmas RRSIG que autentican cada conjunto de recursos, registros DNSKEY que publican las claves públicas de la zona, y registros NSEC3 que previenen la enumeración de la zona (zone walking).

La implementación mantiene la separación de funciones entre la KSK y la ZSK recomendada por el RFC 6841. Esta separación permite rotar la ZSK con frecuencia dado que

firma un elevado número de registros sin necesidad de modificar el ancla de confianza configurada en el resolutor, ya que dicha ancla corresponde a la KSK. La KSK, por su parte, se rota con una periodicidad mucho menor, lo que reduce la complejidad operativa de la gestión de claves y minimiza el riesgo de interrupciones en la cadena de confianza.

El proceso de firmado generó adicionalmente el registro DS (Delegation Signer) de la zona, derivado de la KSK (identificador 28475) mediante el algoritmo de digest SHA-256 (tipo 2). En una jerarquía DNS pública, este registro se publicaría en la zona padre para enlazar la cadena de confianza. Dado que empresa.local es una zona privada sin delegación pública, el registro DS no se publica en un padre; la confianza se establece directamente mediante el ancla configurada en el resolutor. No obstante, la generación del DS confirma que la implementación es compatible con una futura integración en una jerarquía DNSSEC completa.

Figura 29

Registro DS de la zona empresa.local.

```
[steve@steve-elektrodell ~]$ docker exec $AUTORITATIVO cat /etc/bind/zones/dsset-empresa.l
ocal.
empresa.local.        IN DS 28475 13 2 AD598C9D1D5B5E4FDEA4C1FB557959C155F31920D7A65A541
```

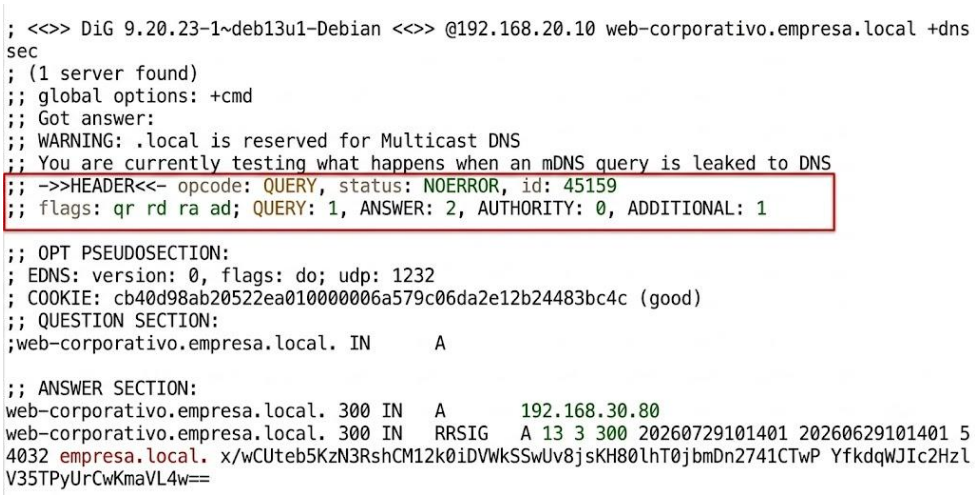
Como se muestra en la Figura 29 el Registro Delegation Signer (DS) derivado de la KSK (identificador 28475) mediante SHA-256. Representa el enlace criptográfico que permitiría delegar la confianza desde la zona padre.

La verificación de que la validación DNSSEC opera correctamente de extremo a extremo se realiza mediante una consulta dig con el flag +dnssec dirigida al servidor recursivo. Una respuesta validada presenta tres indicadores: el flag AD (Authenticated Data) en la cabecera, la presencia de registros RRSIG en la sección de respuesta, y un tamaño de paquete mayor (211 bytes o más en respuestas firmadas, frente a 99–102 bytes sin DNSSEC, pudiendo alcanzar 836 bytes en

respuestas con múltiples registros de firma). La aparición del flag AD confirma que el recursivo validó exitosamente la cadena de firmas contra el ancla de confianza.

Figura 30

Verificación de la validación DNSSEC (flag AD).



Como se muestra la salida del comando dig +dnssec en la que la cabecera muestra el flag AD (Authenticated Data) y la sección de respuesta incluye el registro RRSIG, confirmando la validación criptográfica exitosa por parte del servidor recursivo.

3.5.8. *Módulo de ataque implementado.*

La evaluación de seguridad de esta investigación no reproduce el ataque de envenenamiento de caché por carrera de identificador descrito por Kamisnky (CVE-2008-1447) sino que implementa dos técnicas deterministas y reproducibles para probar la validación criptográfica de DNSSEC: una suplantación de servidor autoritativo (spoofing) mediante el secuestro de la configuración de reenvío del resolutor, y una denegación de servicio por saturación (flooding) mediante carga adversaria sostenida. o de identificadores dentro de esa ventana.

La elección de este enfoque responde a una limitación técnica del ataque clásico. El ataque de Kaminsky explota una condición de carrera en la que un atacante off-path, sin observar el tráfico entre el recursivo y el autoritativo, intenta adivinar el identificador de transacción de 16 bits y el

puerto de origen para inyectar una respuesta falsificada antes de que llegue la legitima. En un resolutor operando en modo forward como el implementado, esta carrera no puede reproducirse de forma consistente: el recursivo no procesa referencias iterativas, la ventana temporal explotable es mínima, BIND 9.20 aleatoriza el puerto de origen, y las herramientas disponibles no alcanzan la tasa de inyección necesaria para cubrir el espacio de identificadores dentro de esa ventana. Por ello, en lugar de perseguir un vector de inyección inviable, se aíslan de forma controlada las dos condiciones que DNSSEC está diseñado para mitigar la aceptación de datos de origen no verificado y la degradación del servicio bajo carga y se evalúa la respuesta del sistema ante cada una.

El componente cuantitativo (E3a y E4a) implementa una denegación de servicio por saturación (flooding): somete al servidor recursivo a una carga adversaria consistente en un flujo sostenido de consultas a nombres reales de la zona empresa.local. Cada consulta obliga al recursivo a resolver a través del reenvío, congestionando la ruta de resolución y permitiendo medir el impacto del ataque sobre la latencia, el throughput y el consumo de recursos, con y sin DNSSEC activo.

El componente cualitativo (E3b y E4b) implementa una suplantación de servidor autoritativo (spoofing) mediante secuestro de configuración: se redirige la directiva de reenvío (forwarders) del recursivo hacia el nodo atacante (192.168.10.66), que actúa como autoritativo malicioso y devuelve, para el dominio web-corporativo.empresa.local, su propia dirección (192.168.10.66) en lugar de la legítima (192.168.30.80). Este mecanismo modela una amenaza de configuración comprometida o de adversario interno con control sobre el resolutor, no una intercepción en la capa de enlace. Con la validación DNSSEC desactivada (E3b), el recursivo acepta y entrega la dirección forjada, con un TTL de 60 segundos; con la validación activa (E4b), la rechaza con código SERVFAIL al no poder verificar las firmas contra el ancla de confianza.

Conviene recalcar que este modelo no reivindica un envenenamiento off-path exitoso por carrera de identificadores, sino que demuestra la propiedad de validación de DNSSEC de forma determinista: ante una respuesta sin firmas válidas encadenadas al ancla de confianza, el recursivo la rechaza. El ataque queda evidenciado por su efecto la aceptación de un registro forjado en ausencia de DNSSEC (E3b), que es precisamente la condición que DNSSEC previene.

El servidor malicioso se implementó en el script dns_falso.py, desarrollado con la biblioteca Scapy. Emplea un socket UDP real enlazado al puerto 53 para evitar el rechazo del kernel, y construye una respuesta con la dirección del atacante, pero sin registros de firma RRSIG, que es justamente la ausencia que DNSSEC detecta.

Bloque de código 14

Fragmento de dns_falso.py.

```
# Construcción de la respuesta forjada (sin firmar)
def construir_respuesta(data):
    dns = DNS(data)                    # parsear la consulta
    if dns.qr != 0 or dns.qdcount < 1:
        return None
    qname = dns[DNSQR].qname.decode()
    if ZONA not in qname or dns[DNSQR].qtype != 1:
        return None
    # Devolver la IP del atacante en lugar de la legítima
    ip_resp = FORJADO.get(qname, IP_GENERICA)   # 192.168.10.66
    resp = DNS(id=dns.id, qr=1, aa=1, ra=0, rcode=0,
               qd=dns.qd,
               an=DNSRR(rrname=qname, type="A", ttl=60, rdata=ip_resp))
    return bytes(resp)                 # NO incluye RRSIG: DNSSEC lo detecta
```

"La generación de la carga adversaria se realizó mediante la herramienta dnsperf ejecutada desde el nodo atacante, produciendo un flujo sostenido de consultas contra el servidor recursivo (aproximadamente 1 240 consultas por segundo sin DNSSEC y 1 120 con DNSSEC), tal como se describe en el componente cuantitativo.

3.5.9. *Herramientas de medición*

Para la recolección de las métricas se emplearon herramientas de código abierto ampliamente utilizadas en el diagnóstico de servicios DNS, complementadas con scripts propios que automatizan su ejecución. La latencia se midió con el comando dig, consultando un conjunto de 30 dominios representativos de la zona empresa.local y extrayendo el tiempo de consulta (Query time) de cada resolución. Para el estado de caché frío, el servicio named del recursivo se reiniciaba antes de cada corrida y se esperaban unos segundos antes de consultar, ya que de lo contrario aparecían errores de conexión; la latencia caliente se obtuvo repitiendo las mismas consultas con la caché ya poblada.

El throughput se evaluó con dnsperf desde el nodo analizador y también desde el Cliente-1-1, ubicado uno en el mismo segmento que el servidor recursivo y otro en el segmento LAN, aplicando cargas de 250, 500, 1000 , 2000 y 4000 consultas por segundo durante intervalos de 120 segundos y registrando el caudal real y la tasa de pérdida. La carga adversaria de los escenarios E3 y E4 se generó con el script generar_ataque.sh, detallado en el Anexo C, que produce un flujo de consultas a nombres existentes de la zona. El consumo de CPU y memoria del recursivo se capturó con docker stats desde el host, los porcentajes reportados por docker stats son relativos a la capacidad total del host (20 núcleos) y no se impuso límite de cgroup a los contenedores; un valor de 8.9 % equivale aproximadamente a 1.8 núcleos y el tráfico DNS se registró y analizó con Wireshark sobre la interfaz del servidor recursivo, lo que permitió evidenciar el tamaño de las respuestas firmadas y el contraste entre las respuestas NOERROR y SERVFAIL.

Sobre estas herramientas se desarrollaron los scripts detallados en el Anexo D: medir.sh, que automatiza las mediciones de latencia (fría y caliente), throughput y consumo de recursos, almacenando los resultados por escenario; y verificar-cache.sh, que monitorea en tiempo real el

estado de la caché del recursivo, distinguiendo si la dirección devuelta corresponde al servidor legítimo (192.168.30.80), al atacante (192.168.10.66) o si la respuesta se rechaza con código SERVFAIL.

3.6 Ejecución de pruebas

3.6.1. *Escenarios experimentales*

Se definieron cuatro escenarios de prueba que combinan la presencia o ausencia de DNSSEC con la ejecución del ataque de envenenamiento de caché, permitiendo evaluar tanto el rendimiento base como la efectividad de las medidas de protección.

La Tabla 16 presenta la matriz de configuración de cada escenario, especificando las variables activas en cada uno.

Tabla 16

Escenarios experimentales definidos.

Esc.	**DNSSEC**	**Vector de ataque**	**Tráfico de fondo**	**Propósito de medición**
E1	Desactivado	Ninguno	Sí	Línea base de rendimiento (latencia / throughput)
E2	Activado	Ninguno	Sí	Sobrecosto (overhead) de DNSSEC en operación normal

E3	Desactivado	Carga adversaria + respuesta forjada	Sí	Rendimiento bajo carga y aceptación de la respuesta forjada
E4	Activado	Carga adversaria + respuesta forjada	Sí	Rendimiento bajo carga y rechazo de la respuesta forjada (SERVFAIL)

Los escenarios E1 y E2 establecen el comportamiento del servicio en operación normal, permitiendo cuantificar el sobrecosto introducido por DNSSEC. Los escenarios E3 y E4 evalúan directamente la efectividad del mecanismo de protección frente a los ataques antes expuestos, contrastando la infraestructura sin protección con la protegida ante el mismo vector.

3.6.2. *Métricas evaluadas*

Latencia que es el tiempo transcurrido desde el envío de la consulta DNS hasta la recepción de la respuesta, medido en milisegundos. Las mediciones se obtuvieron ejecutando de forma repetida el comando dig sobre el conjunto de dominios de prueba, recolectando el tiempo de consulta (Query time) reportado en cada ejecución; sobre ese conjunto de muestras se calcularon los estadísticos descriptivos indicados. Se distinguen dos estados: caché frío (primera resolución tras reiniciar el servicio) y caché caliente (resolución desde memoria). El número de muestras difiere entre ambos estados debido al propio procedimiento: cada medición en caché frío exige vaciar previamente la caché del resolutor mediante rndc flush, lo que limita el número de

repeticiones factibles, mientras que el estado de caché caliente permite acumular un mayor número de mediciones en el mismo intervalo de tiempo.

Throughput para la capacidad de procesamiento del servidor recursivo, expresada en consultas por segundo (QPS), evaluada mediante dnsperf a cargas de 250,500,1000,2000 y 4000 durante intervalos de 120 segundos cada uno, registrando el throughput real alcanzado y la tasa de pérdida.

Uso de recursos que se toma en cuenta el consumo de CPU y memoria RAM del servidor recursivo, capturado mediante docker stats que son relativos a la capacidad del host, dando un valor de medida de porcentaje como ejempo de 8.9 % a 1.8 núcleos, desde el host durante la ejecución de cada escenario.

La efectividad de validación que se mide con el comportamiento del recursivo ante la respuesta forjada: aceptación (respuesta NOERROR con la dirección del atacante, 192.168.10.66) o rechazo (respuesta SERVFAIL). Esta métrica es de naturaleza cualitativa y binaria, y constituye la evidencia funcional de la protección de DNSSEC.

3.6.3. *Procedimiento de Medición de Escenarios*

Para garantizar la comparación entre escenarios, cada corrida siguió el mismo procedimiento. Primero se inicializaron todos los nodos con su direccionamiento IP y sus servicios DNS, y se comprobó la conectividad entre zonas mediante pruebas de ping y la resolución correcta con una consulta dig desde el cliente. A continuación, se puso en marcha el tráfico de fondo con trafico-continuo.sh, que simula la carga de varios usuarios y ayuda a poblar la caché antes de medir exceptuando en la medida con caché frio. Sobre esa base se ejecutó la serie de mediciones de latencia, consultando dominios representativos de la zona empresa.local tanto en caché frío como caliente; para el estado frío se reiniciaba el servicio named del recursivo antes de cada corrida y se

esperaban unos segundos antes de consultar, ya que de lo contrario aparecían errores de conexión. Después se realizaron las pruebas de throughput con dnsperf, se registró el consumo de recursos con docker stats. Todos los resultados se guardaron en archivos CSV, TXT y PCAP para su análisis posterior. En los escenarios con ataque se añadió un paso final, lanzar la carga adversaria desde el nodo atacante.

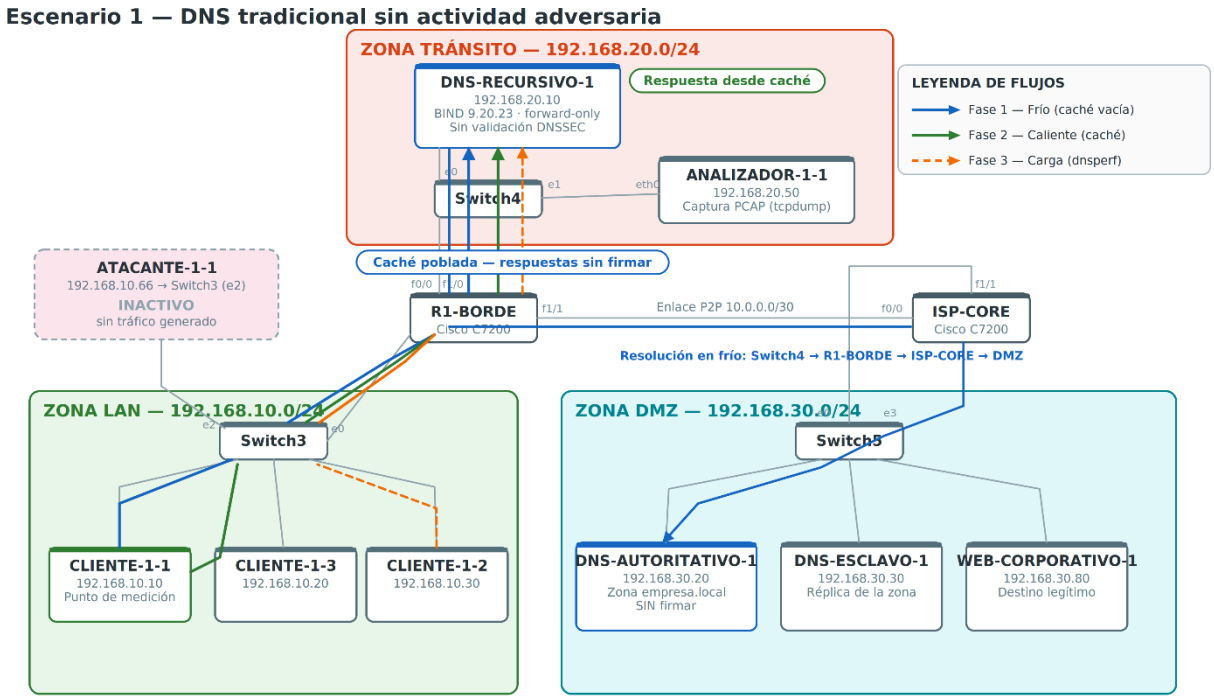
3.7 Resultados experimentales

3.7.1. *Escenario 1 - DNS tradicional sin ataque (línea base)*

El primer escenario se estableció como la línea base de rendimiento del DNS tradicional sin extensiones de seguridad y sin interferencia de atacantes. Las mediciones se ejecutaron en dos fases: sin tráfico de fondo (cuando no hay información en la caché), con tráfico de fondo (cuando ya hay información en la caché). Se distribuyeron las consultas aleatoriamente entre dominios configurados en la zona empresa.local.

Figura 31

Escenario 1 comportamiento normal de la infraestructura frente a distintos estados de la caché del servidor recursivo



.

El diagrama ilustra la trayectoria de los paquetes DNS a través de la topología simulada durante las dos fases de evaluación planificada. Las trazas continuas azules (Fase 1) representan la resolución en frío, es decir, buscamos primero en el Servidor Autoritativo, evidenciando el enrutamiento iterativo de las consultas originadas en la ZONA LAN, pasando por el servidor recursivo en la ZONA TRÁNSITO, hasta alcanzar el servidor autoritativo en la ZONA DMZ. Las líneas continuas verdes (Fase 2) detallan la resolución en caliente, donde las respuestas se sirven directamente desde la caché poblada del servidor recursivo hacia los clientes, absorbiendo la carga concurrente. Las trazas discontinuas naranjas (Fase 3) muestran el patrón de tráfico durante el periodo de estabilización y recuperación de recursos post-carga. Como un punto importante es

que el nodo del Atacante no está activo puesto que primero se mide en un escenario sin ninguna actividad maliciosa y aislamiento de la medición de línea base.

3.7.2. *Configuración del Escenario*

Para la configuración del Escenario 1, se verificó que el servidor recursivo (192.168.20.10) operara sin realizar la validación criptográfica de las respuestas. Esta condición se comprobó ejecutando el comando `grep dnssec-validation /etc/bind/named.conf.options` dentro del contenedor del recursivo, donde la obtención del parámetro dnssec-validation no; confirmó la ausencia de validación de firmas DNSSEC.

Asimismo, de manera previa a la ejecución de cada medición en caché frío, se procedió a vaciar la memoria caché del recursivo mediante el reinicio del servicio DNS utilizando la instrucción `pkill named; sleep 3; named -u bind`.

Dicha secuencia de comandos finaliza el proceso de red, establece una pausa de tres segundos y lo inicializa nuevamente, lo cual asegura que el servicio arranque con la memoria completamente limpia y garantiza que la primera resolución de cada dominio se efectúe obligatoriamente a través de una recursión completa hacia el servidor autoritativo.

Los dos estados de caché evaluados son el caché frío donde el recursivo no contiene los registros; cada consulta obliga a la resolución iterativa hacia el autoritativo. Se mide sin tráfico de fondo, ya que este poblaría la caché. Representa el peor caso de latencia.

Luego está el caché caliente donde los registros ya residen en la caché del recursivo, poblada por el tráfico de fondo. El recursivo responde directamente desde memoria. Representa la operación realista bajo carga sostenida.

Los dos tipos de tráfico son el tráfico de fondo que son consultas continúas generadas desde CLIENTE-1-2 y CLIENTE-1-3, simulando usuarios normales de la empresa. Mantiene la caché

del recursivo caliente durante las mediciones en dicho estado y continuamos con el tráfico de estrés que tiene una carga controlada de consultas por segundo (QPS) generada mediante la herramienta dnsperf, para evaluar el rendimiento del servidor bajo presión.

Un aspecto relevante es el punto desde el cual se mide cada métrica, la latencia se mide desde CLIENTE-1-1 (192.168.10.10), de modo que el valor obtenido refleje la ruta real que recorre un usuario: cliente → switch de la LAN → router R1-BORDE → servidor recursivo. Medir la latencia desde un nodo situado en la misma red del recursivo subestimaría el valor, al no atravesar los equipos de enrutamiento.

El consumo de CPU y RAM se captura desde el host mediante docker stats, con la información proporcionada por el motor de contenedores. Los datos de latencia y estado de cada consulta se registran en archivos CSV; los valores de throughput se obtienen de la salida de dnsperf; y el consumo de recursos se almacena en un archivo con marca de tiempo. Todos los resultados se respaldan tanto en la carpeta persistente del contenedor como en el host, para evitar su pérdida ante reinicios del entorno. Tomando estos pasos podemos obtener los resultados que se muestran en la Tabla 17 que está a continuación.

Tabla 17

Resultados del Escenario 1 - DNS tradicional sin ataque.

Condición	**N**	**OK**	**Latencia prom.**	**Mín**	**Máx**	**Std**	**Varianza**	**IC**
Caché frío*	267	100 %	33.55 ms	10 ms	50 ms	15.10 ms	228.01 ms^2	[30.41, 36.69] ms

Caché caliente**	267	100 %	14.70 ms	5 ms	20 ms	4.12 ms	16.97 ms^2	[14.21 , 15.19] ms

Nota: Caché frío: primera ronda de consultas con la caché del recursivo vacía; cada consulta se resuelve hasta el servidor autoritativo.*

*** Caché caliente: consultas resueltas desde la caché del recursivo, con tráfico de fondo generado por el cliente 2.*

En la condición de caché frío, donde cada consulta debe resolverse recorriendo la cadena completa hasta el servidor autoritativo y retornando por el recursivo, la latencia promedio fue de 33.55 ms, con un intervalo de confianza al 95 % de [30.41, 36.69] ms y una varianza de 228.01 ms^2. Una vez poblada la caché del recursivo, la condición de caché caliente redujo la latencia promedio a 14.70 ms, con un intervalo de confianza al 95 % de [14.21, 15.19] ms y una varianza de 16.97 ms^2, lo que representa una disminución cercana al 56 % respecto a la condición fría. La marcada reducción en la desviación estándar (de 15.10 ms a 4.12 ms) refleja además una mayor estabilidad en las respuestas servidas desde caché, al eliminarse la variabilidad asociada al recorrido hacia el autoritativo. Dado que los intervalos de confianza de ambas condiciones no se solapan, la diferencia entre resolución en frío y en caliente es estadísticamente significativa. Cabe precisar que esta mejora es atribuible al mecanismo de almacenamiento en caché del recursivo y no al tráfico de fondo generado por el cliente 2, el cual se incorporó únicamente como condición de realismo operativo y no como factor de optimización de la latencia.

Figura 32

Resolución en E1: la respuesta incluye RRSIG pero el recursivo no valida (ausencia del flag AD)

```
; <<>> DiG 9.20.23-1~deb13u1-Debian <<>> @192.168.20.10 web-corporativo.empresa.local A +dnssec
; (1 server found)
;; global options: +cmd
;; Got answer:
;; WARNING: .local is reserved for Multicast DNS
;; You are currently testing what happens when an mDNS query is leaked to DNS
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6315
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do: udp: 1232
; COOKIE: 643b2f34a256b579010000000a474380b66cadd0e49d9b7f (good)
;; QUESTION SECTION:
;web-corporativo.empresa.local. IN      A

;; ANSWER SECTION:
web-corporativo.empresa.local. 16 IN    A       192.168.30.80
web-corporativo.empresa.local. 16 IN    RRSIG   A 13 3 300 20260729101401 20266629101401 54032 empresa.local.

;; Query time: 19 msec
;; SERVER: 192.168.20.10#53(192.168.20.10) (UDP)
;; WHEN: Fri Jul 03 00:07:12 -05 2026
;; MSG SIZE  rcvd: 211
```

Como se muestra en la Figura 32 podemos observar que no se encuentra la Flag que identifica de manera adecuada la verificación del DNS con la extensión de seguridad, pero en el caso del Escenario 1 esto no es autenticado para probar la línea base de la evaluación en el escenario mencionado.

Figura 33

Comparación de latencia y throughput entre caché frío y caliente en E1.

```
echo "########## E1 CACHE CALIENTE ##########"
bash /tmp/analizar.sh e1 caliente
########## E1 CACHÉ FRÍO ##########
======== RESUMEN E1 | cache FRÍO ========
--- LATENCIA (ms) ---
  Consultas: 267
  Promedio: 33.55 ms
  Mínimo: 10 ms
  Máximo: 50 ms
  Desv.est: 15.10 ms
--- ESTADOS ---
       267 ok
--- THROUGHPUT ---
  100 QPS -> 100.0 q/s | completadas:2000 | perdidas:0
  150 QPS -> 150.0 q/s | completadas:3000 | perdidas:0
  300 QPS -> 300.0 q/s | completadas:6000 | perdidas:0
================================================

########## E1 CACHE CALIENTE ##########
======== RESUMEN E1 | cache CALIENTE ========
--- LATENCIA (ms) ---
  Consultas: 267
  Promedio: 14.70 ms
  Mínimo: 0 ms
  Máximo: 20 ms
  Desv.est: 4.12 ms
--- ESTADOS ---
       267 ok
```

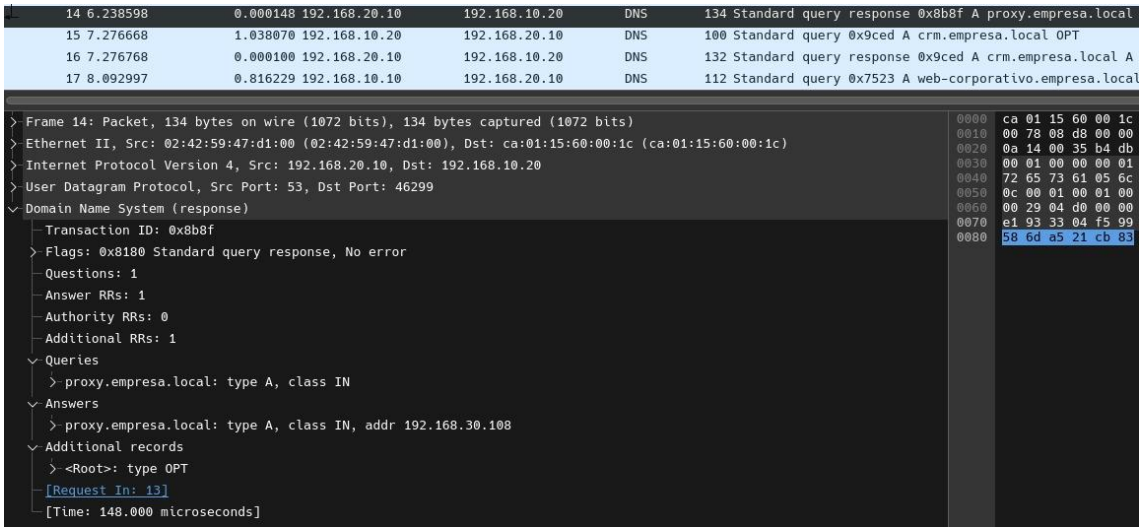
Como se muestra en la Figura 33 de caché frío, la latencia promedio fue de 33.55 ms, correspondiente a la resolución iterativa hacia el servidor autoritativo cuando los dominios aún no residen en la caché del recursivo.

Una vez que la caché está cargada mediante el tráfico de fondo (caché caliente), la latencia promedio descendió a 14.70 ms, notablemente más estable. En este estado el recursivo responde desde su memoria local sin recurrir al autoritativo, lo que reduce la latencia respecto al caché frío.

Es importante recalcar que los valores de latencia obtenidos corresponden a un entorno virtualizado en GNS3, sin la latencia física de una red de producción.

Figura 34

Análisis de PCAP en el Servidor Recursivo.



Se capturó el tráfico DNS en el servidor recursivo mediante tcpdump, filtrando el puerto 53. La captura se realiza en el recursivo porque, al tratarse de una red conmutada, es el punto por el que necesariamente transita todo el tráfico de consultas y respuestas.

Es importante señalar que, dado que la zona empresa.local se encuentra firmada, el registro RRSIG aparece en la respuesta siempre que el cliente lo solicite, incluso en el Escenario 1. Sin embargo, en la cabecera de la respuesta no se presenta el flag AD (Authenticated Data), lo que

indica que el servidor recursivo entrega la firma, pero no la valida. Esta ausencia del flag AD es la característica distintiva del Escenario 1: existe información de firma, pero no autenticación por parte del resolutor.

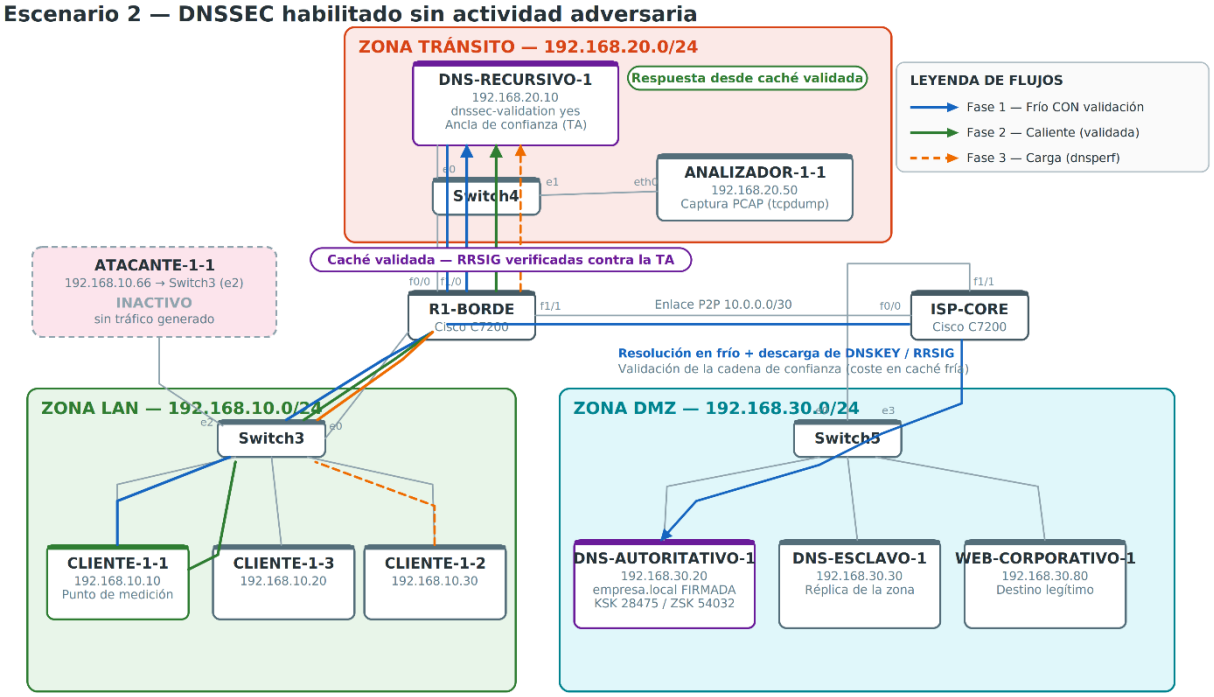
3.7.3. *Escenario 2 - DNS con DNSSEC sin ataque*

El segundo escenario evaluó el impacto de DNSSEC en el rendimiento del servidor DNS. La zona empresa.local fue firmada con claves ECDSAP256SHA256, generando 371 firmas RRSIG para los 90 dominios configurados, y el servidor recursivo fue configurado con validación DNSSEC habilitada y el trust anchor correspondiente a la KSK del autoritativo.

En el contexto de esta evaluación, se denomina overhead (sobrecosto) al incremento porcentual en la latencia de resolución introducido por las operaciones criptográficas de DNSSEC respecto al DNS tradicional, calculado como la diferencia relativa entre la latencia con DNSSEC activo y la latencia base. Este sobrecosto debe interpretarse en función del estado de la caché del servidor recursivo. En una resolución con caché frío la primera consulta a un dominio, el recursivo debe descargar y validar los registros DNSKEY y las firmas RRSIG, lo que concentra el mayor costo. En cambio, en las consultas con caché caliente resoluciones subsecuentes, una vez validadas y almacenadas las claves, el sobrecosto se reduce de forma drástica, llegando a ser prácticamente nulo. Por esta razón, las mediciones que se presentan a continuación se reportan diferenciando ambas condiciones.

Figura 35

Escenario 2 Validación DNSSEC Habilitada y Ancla de Confianza configurada



La figura 35 proporciona una visualización detallada de la topología de red utilizada para el Escenario 2, centrado en el impacto de DNSSEC en la zona firmada "empresa.local". El flujo detallado de paquetes para la resolución en frío (línea azul), la resolución en caliente (línea verde) y la estabilidad post-carga (línea naranja) se detalla claramente en la parte de paquetes.

El nodo Atacante, ATACANTE-1-1, en el cuadro rojo, está etiquetado como con su amenaza mitigada por la validación DNSSEC y está visualmente inactivo, con la línea de conexión en estado no funcional y sin iconos de actividad, confirmando que el atacante permanece inactivo durante esta fase de medición. Un nodo de análisis, Analizador-1-1, está presente para monitorear el tráfico validado.

3.7.4. *Configuración del Escenario 2*

A diferencia del Escenario 1, aquí el servidor recursivo valida criptográficamente cada respuesta. La activación de DNSSEC requiere dos elementos, activar la validación y proporcionar al recursivo el ancla de confianza (trust anchor).

El trust anchor es la parte pública de la Key Signing Key (KSK) de la zona empresa.local, alojada en el servidor autoritativo. Con ella, el recursivo puede verificar la cadena de confianza: la KSK firma la Zone Signing Key (ZSK), la ZSK firma los registros de la zona, y por tanto las respuestas son auténticas. La clave pública KSK se identifica por el flag 257 (frente al 256 de la ZSK) y utiliza el algoritmo 13 (ECDSAP256SHA256).

El trust anchor se declara en el recursivo mediante un bloque trust-anchors con la directiva static-key, apropiada para zonas privadas que no emplean la actualización automática de claves definida en el RFC 5011.};

Posteriormente se activa la validación cambiando en named.conf.options la directiva dnssec-validation no; por dnssec-validation yes. Antes de reiniciar el servicio se verifica la sintaxis con named-checkconf para evitar dejar el servidor DNS inoperativo por un error de configuración.

La verificación de que DNSSEC quedó activo se realiza observando la cabecera de la respuesta a una consulta con el bit DNSSEC habilitado, dig @192.168.20.10 web-corporativo.empresa.local A +dnssec

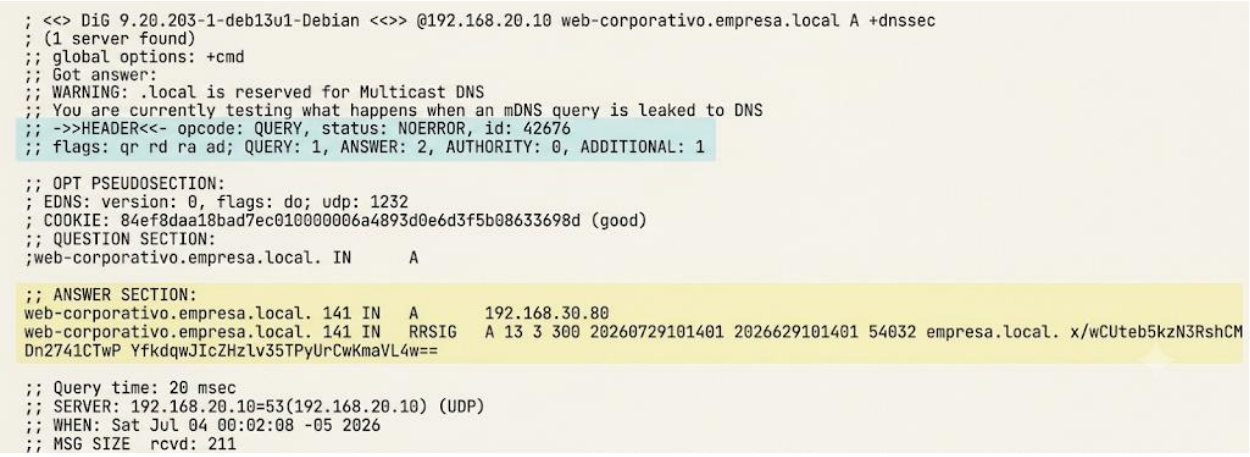
La aparición de la flag AD (Authenticated Data) en la línea de flags (flags: qr rd ra ad) confirma que el recursivo valida las firmas. Su ausencia caracterizaba al Escenario 1; su presencia caracteriza al Escenario 2.

3.7.5. *Procedimiento de Medición del Escenario 2*

El procedimiento es idéntico al del Escenario 1, para garantizar que las diferencias observadas se deban únicamente a la activación de DNSSEC y no a variaciones en el método.

Figura 36

Verificación de DNSSEC activo en E2



En la Figura 36 se muestra la consulta dig +dnssec desde CLIENTE-1-1 al recursivo (192.168.20.10) para web-corporativo.empresa.local. El encabezado presenta status: NOERROR y el flag ad (authenticated data), que confirma la validación criptográfica exitosa. La sección de respuesta incluye el registro A legítimo (192.168.30.80) junto a su firma RRSIG (algoritmo 13, ECDSAP256SHA256, keytag 54032), y el tamaño de respuesta asciende a 211 bytes por la inclusión de la firma. La presencia del flag ad en EDNS evidencia que la consulta solicitó los registros DNSSEC.

Figura 37

Comparación de latencia entre caché frío y caliente en E2.

```
       INFORME DE RENDIMIENTO DE CACHÉ - PRUEBA E2
================================================
======== RESUMEN E2 | CACHÉ FRÍO ========
--- LATENCIA (ms) ---
  Promedio:       42.93 ms
  Máximo:         59 ms
  Desv. Estándar: 10.59 ms
--- ESTADOS ---
      89 validada dnssec
--- THROUGHPUT ---
  Carga (QPS) |  Tasa (q/s) | Completadas |   Perdidas
          100 |       100.0 |        2000 |          0
          150 |       150.0 |        3000 |          0
          300 |       300.0 |        6000 |          0
================================================
======== RESUMEN E2 | CACHÉ CALIENTE ========
--- LATENCIA (ms) ---
  Consultas: 267
  Promedio: 15.09 ms
  Mínimo: 9 ms
  Máximo: 20 ms
  Desv. Estándar: 3.62 ms
--- ESTADOS ---
     267 validada dnssec
```

En condiciones normales, el sobrecosto de DNSSEC se concentra en la primera resolución (caché frío), donde la latencia aumenta de 33.55 ms a 42.93 ms (+28 %) debido a la descarga y validación inicial de los registros DNSKEY y RRSIG. Una vez poblada la caché de claves, la latencia en caché caliente es prácticamente idéntica (14.70 ms sin DNSSEC frente a 15.09 ms con DNSSEC, +2 %), por lo que el overhead resulta despreciable en consultas concurrentes.

Figura 38

Medición bajo tráfico concurrente con DNSSEC activo y rendimiento estable

```
root@CLIENTE-1-1:/# sleep 30
echo "=== E2 CON TRÁFICO 300 ==="
/opt/scripts/medir-avanzado.sh con_dnssec_con_trafico_300 300

echo "=== E2 CON TRÁFICO 500 ==="
/opt/scripts/medir-avanzado.sh con_dnssec_con_trafico_500 500
=== E2 CON TRÁFICO 500 ===
==================================================
  MEDICIÓN - Escenario: con_dnssec_con_trafico_300 | 300 consultas |
==================================================
 [10%] 30/300 - OK:30 Env:0 Fail:0 Prom:27ms
 [20%] 60/300 - OK:60 Env:0 Fail:0 Prom:27ms
 [30%] 90/300 - OK:90 Env:0 Fail:0 Prom:27ms
 [40%] 120/300 - OK:120 Env:0 Fail:0 Prom:27ms
 [50%] 150/300 - OK:150 Env:0 Fail:0 Prom:27ms
 [60%] 180/300 - OK:180 Env:0 Fail:0 Prom:27ms
 [70%] 210/300 - OK:210 Env:0 Fail:0 Prom:27ms
 [80%] 240/300 - OK:240 Env:0 Fail:0 Prom:27ms
 [90%] 270/300 - OK:270 Env:0 Fail:0 Prom:27ms
 [100%] 300/300 - OK:300 Env:0 Fail:0 Prom:27ms
==================================================
  RESULTADOS - con_dnssec_con_trafico_300
  Consultas: 300 | OK: 300 | Envenenadas: 0 | Fallidas: 0
  Latencia promedio: 27 ms
  CSV: /tmp/latencia_con_dnssec_con_trafico_300.csv
==================================================
=== E2 CON TRÁFICO 500 ===
==================================================
  MEDICIÓN - Escenario: con_dnssec_con_trafico_500 | 300 consultas |
==================================================
 [10%] 50/500 - OK:10 Env:0 Fail:0 Prom:27ms
 [20%] 100/500 - OK:100 Env:0 Fail:0 Prom:27ms
 [30%] 150/500 - OK:150 Env:0 Fail:0 Prom:27ms
 [40%] 200/500 - OK:200 Env:0 Fail:0 Prom:27ms
 [50%] 250/500 - OK:250 Env:0 Fail:0 Prom:27ms
 [60%] 300/500 - OK:350 Env:0 Fail:0 Prom:27ms
 [70%] 350/500 - OK:350 Env:0 Fail:0 Prom:27ms
 [80%] 400/500 - OK:400 Env:0 Fail:0 Prom:28ms
 [90%] 450/500 - OK:450 Env:0 Fail:0 Prom:32ms
 [100%] 500/500 - OK:500 Env:0 Fail:0 Prom:32ms
==================================================
  RESULTADOS - con_dnssec_con_trafico_500
  Consultas: 500 | OK: 500 | Envenenadas: 0 | Fallidas: 0
==================================================
```

Se muestra la ejecución del script de medición para E2 con carga de tráfico concurrente a 300 y 500 consultas. En ambos niveles la totalidad de las consultas se resolvió correctamente con latencia promedio estable en torno a 27 ms, que asciende ligeramente a 32 ms en el tramo final de la corrida de 500. El resultado evidencia que DNSSEC sostiene el servicio sin pérdidas ni degradación apreciable bajo tráfico concurrente.

Tabla 18

Uso de recursos del sistema con DNSSEC activo.

Contenedor (Nodo GNS3)	**Consumo de CPU del contenedor (Promedio)**	**Consumo de RAM del contenedor (Aprox.)**
CLIENTE-1-1	~ 11.91 %	6.5 MiB
CLIENTE-1-2	~ 3.65 %	5.5 MiB
Servidor-DNS-RECURSIVO-1	~ 0.15 %	211.0 MiB

Estos resultados demuestran que el impacto de DNSSEC en el rendimiento es significativamente menor de lo esperado cuando el mecanismo de caché está activo, ya que en operación sostenida con caché poblada el sobrecosto resulta despreciable y solo se manifiesta en la primera resolución. Este costo es aceptable para entornos corporativos, considerando que DNSSEC proporciona protección criptográfica contra el envenenamiento de caché.

A diferencia de mediciones con caché TTL forzado a 1 segundo (que pueden mostrar overheads superiores al 90%), estos resultados reflejan el comportamiento real de DNSSEC en un entorno operativo donde el caché mitiga significativamente el costo de la validación criptográfica. El consumo de recursos se mantuvo prácticamente idéntico, con apenas 7 MB adicionales de RAM en el servidor recursivo para almacenar las claves y firmas validadas.

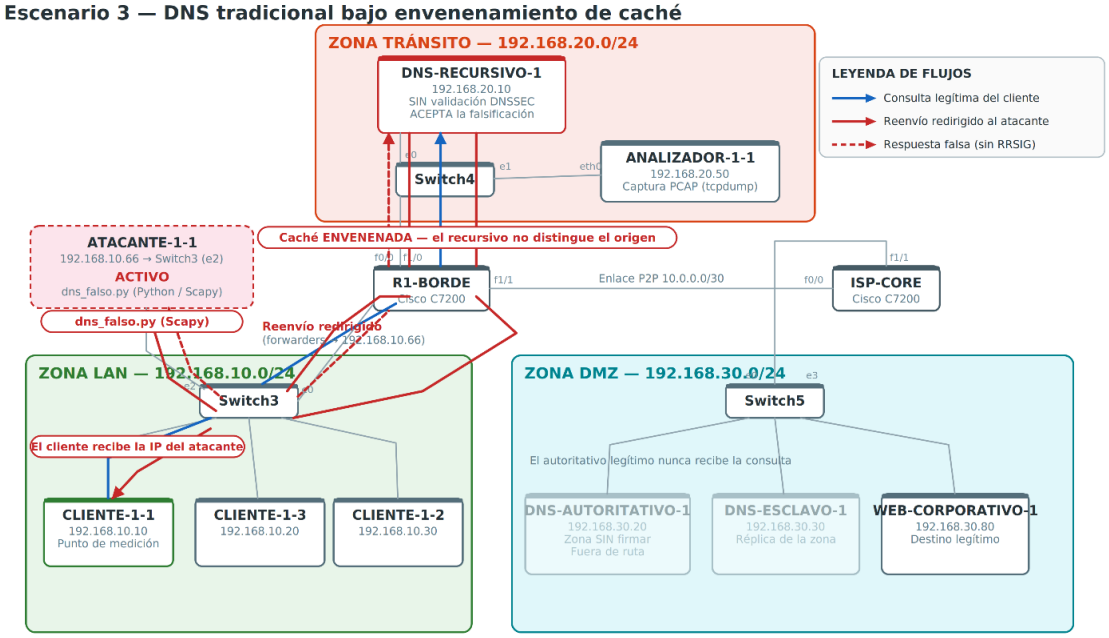
3.7.6. *Escenario 3 - DNS tradicional bajo carga Adversaria*

El Escenario 3 evalúa el comportamiento del servicio DNS sin validación DNSSEC sometido a una carga adversaria. El escenario se divide en dos componentes: E3a, de carácter cuantitativo, que mide el impacto de la carga sobre el rendimiento; y E3b, de carácter cualitativo, que demuestra la aceptación de una respuesta forjada en ausencia de validación. Los subescenarios

a (rendimiento) mantienen el reenvío legítimo bajo Flooding, mientras que los subescenarios b (validación) redirigen el reenvío al nodo malicioso.

Figura 39

Explotación de vulnerabilidad de protocolo mediante carga adversaria en DNS tradicional (Escenario 3).



La Figura 39 muestra la carga adversaria dirigida al Servidor-DNS-RECURSIVO-1, ubicado en la ZONA TRANSIT. El nodo ATACANTE-1-1 inyecta un flujo masivo de consultas (trazas rojas discontinuas) hacia el recursivo, generando contención de recursos. La vulnerabilidad estructural del DNS tradicional se materializa por la ausencia de validación criptográfica del origen de los datos: el resolutor acepta cualquier respuesta sintácticamente válida sin verificar su procedencia.

Este escenario no reproduce el ataque de Kaminsky, sino que implementa una suplantación determinista del servidor autoritativo (spoofing) mediante la redirección del reenvío del resolutor. El mecanismo a nivel de red es el siguiente: el servidor recursivo de la zona TRANSIT no acepta

de forma espontánea que un host de la LAN actúe como su autoritativo; la redirección se produce porque se modifica de manera controlada la directiva forwarders de la zona empresa.local en la configuración del recursivo, apuntándola a la dirección del nodo ATACANTE-1-1 (192.168.10.66) en lugar del autoritativo legítimo (192.168.30.20). Este modelo representa una amenaza de configuración comprometida o de adversario interno con control sobre el resolutor, no una intercepción en la capa de enlace (ARP spoofing). El objetivo del escenario no es demostrar la viabilidad del vector de inyección, sino evaluar de forma determinista si DNSSEC rechaza una respuesta forjada, propiedad de seguridad que es independiente del mecanismo por el cual la respuesta falsa llega al resolutor.

3.7.7. *Escenario E3a – Rendimiento sin DNSSEC bajo carga adversaria.*

Bajo la carga adversaria, la latencia en caché caliente se elevó de 14.70 ms (E1, sin ataque) a 66 ms, y el consumo de CPU del recursivo pasó de 0.56 % a 8.9 %, evidenciando la contención de recursos provocada por el tráfico.

Tabla 19

Resultados del Escenario 3a DNS sin DNSSEC bajo carga adversaria.

Condición	N	OK	Media	Mediana	Mín	Máx	Std	Varianza	IC 95 %
Caché frío	180	100 %	64.65 ms	63 ms	59 ms	73 ms	3.19 ms	10.17 ms²	[64.18, 65.12] ms
Caché caliente	180	100 %	64.83 ms	66 ms	56 ms	73 ms	2.67 ms	7.12 ms²	[64.44, 65.23] ms

La latencia de resolución se niveló en ambas condiciones de caché. La condición de caché frío registró una latencia media de 64.65 ms (IC 95 %: [64.18, 65.12] ms; varianza 10.17 ms²) y la de caché caliente 64.83 ms (IC 95 %: [64.44, 65.23] ms; varianza 7.12 ms²), valores estadísticamente indistinguibles dado el solapamiento de sus intervalos de confianza. Este comportamiento contrasta con el Escenario 1 sin carga, donde la caché caliente reducía la latencia a menos de la mitad respecto a la fría. La explicación es que el tráfico de inundación satura la cola de procesamiento del servidor recursivo, afectando por igual a las consultas resueltas desde caché y a las que recorren la cadena hasta el autoritativo; en consecuencia, el beneficio del almacenamiento en caché se anula bajo condiciones adversarias. El consumo de CPU del contenedor recursivo durante el ataque se mantuvo en 4.37 % de un núcleo en promedio (máximo 4.89 %), lo que indica que el impacto del flooding se manifiesta en la latencia de encolamiento más que en la saturación de CPU del proceso BIND.

3.7.8. *Escenario E3b-Aceptación de la respuesta forjada sin DNSSEC.*

Para demostrar la aceptación de datos forjados en ausencia de validación, el reenvío del recursivo se redirigió hacia el nodo ATACANTE-1-1 (192.168.10.66), configurado como autoritativo malicioso mediante el script dns_falso.py, que responde a las consultas de la zona empresa.local sin incluir registros de firma RRSIG. Con la validación DNSSEC desactivada, el recursivo aceptó la respuesta forjada y la entregó al cliente. Cabe distinguir que 192.168.30.20 corresponde al servidor autoritativo legítimo de la zona, mientras que 192.168.30.80 es el host del servidor web corporativo al que resuelve el registro A de web-corporativo.empresa.local. El ataque no suplanta al servidor web directamente, sino que altera la resolución del nombre para que apunte a la infraestructura del atacante en lugar de a .30.80.

Figura 40

Verificación del envenenamiento en E3b mediante consultas repetidas.

```
[steve@steve-elektrodell ~]$ for i in $(seq 1 15); do   docker exec "$REC" rndc flush 2>/dev/null;
cal A +short +timeout=5);   echo "[$i] $IP $([ "$IP" = "192.168.10.66" ] && echo "ENVENENADO!" ||
[1] 192.168.10.66 ENVENENADO!
[2] 192.168.10.66 ENVENENADO!
[3] 192.168.10.66 ENVENENADO!
[4] 192.168.10.66 ENVENENADO!
[5] 192.168.10.66 ENVENENADO!
[6] 192.168.10.66 ENVENENADO!
[7] 192.168.10.66 ENVENENADO!
[8] 192.168.10.66 ENVENENADO!
[9] 192.168.10.66 ENVENENADO!
[10] 192.168.10.66 ENVENENADO!
[11] 192.168.10.66 ENVENENADO!
[12] 192.168.10.66 ENVENENADO!
[13] 192.168.10.66 ENVENENADO!
[14] 192.168.10.66 ENVENENADO!
[15] 192.168.10.66 ENVENENADO!
```

La consulta desde CLIENTE-1-1 hacia el recursivo (192.168.20.10) para web-corporativo.empresa.local devolvió la dirección del atacante (192.168.10.66) en lugar de la legítima (192.168.30.80), con código NOERROR y el registro presente en la sección de respuesta (ANSWER: 1). El valor de TTL igual a 60 confirma que el registro forjado quedó almacenado en la caché del recursivo: mientras dure ese TTL, cualquier cliente de la red que consulte el mismo nombre recibirá la dirección falsa, sin que el resolutor verifique en ningún momento la procedencia del dato. La respuesta forjada tuvo un tamaño de 102 bytes, sin registros RRSIG.

Tabla 20

Contraste de validación DNSSEC ante la respuesta forjada (E3b sin DNSSEC / E4b con DNSSEC).

Métrica	**Sin DNSSEC (E3b)**	**Con DNSSEC (E4b)**
Respuesta al cliente	NOERROR	SERVFAIL
Dirección entregada	192.168.10.66 (forjada)	ninguna (rechazada)
TTL del registro	60 s	—*
Tamaño de respuesta	102 bytes	86 bytes

Registro en caché	forjado, aceptado	ninguno
Estado de la caché	comprometido	íntegro

**No aplica. La respuesta SERVFAIL del Escenario 4b no contiene ningún registro de recurso (ANSWER: 0), por lo que no existe un TTL asociado. La ausencia de TTL es consecuencia directa de la validación DNSSEC: al no poder el atacante entregar la firma RRSIG correspondiente, el servidor recursivo rechaza la respuesta completa y no almacena ningún registro en caché.*

Los valores son confirmados mediante la salida de dig y la captura de tráfico (pcap) de cada escenario. En E4b el recursivo recibe la respuesta forjada, intenta validar su firma, no la encuentra y responde SERVFAIL, protegiendo al cliente.

Figura 41

Aceptación de la respuesta forjada en E3b: detalle de la resolución.

```
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 65402
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; ANSWER SECTION:
web-corporativo.empresa.local. 57 IN  A  192.168.10.66

;; Query time: 11 msec
;; SERVER: 192.168.20.10#53
```

El registro forjado en la caché del recursivo tiene una consecuencia directa a nivel de aplicación: cualquier cliente que resuelva web-corporativo.empresa.local sería dirigido a la infraestructura del atacante (192.168.10.66) en lugar del servidor web corporativo legítimo (192.168.30.80), materializando un escenario de suplantación de servicio (phishing) o de denegación de acceso al recurso real.

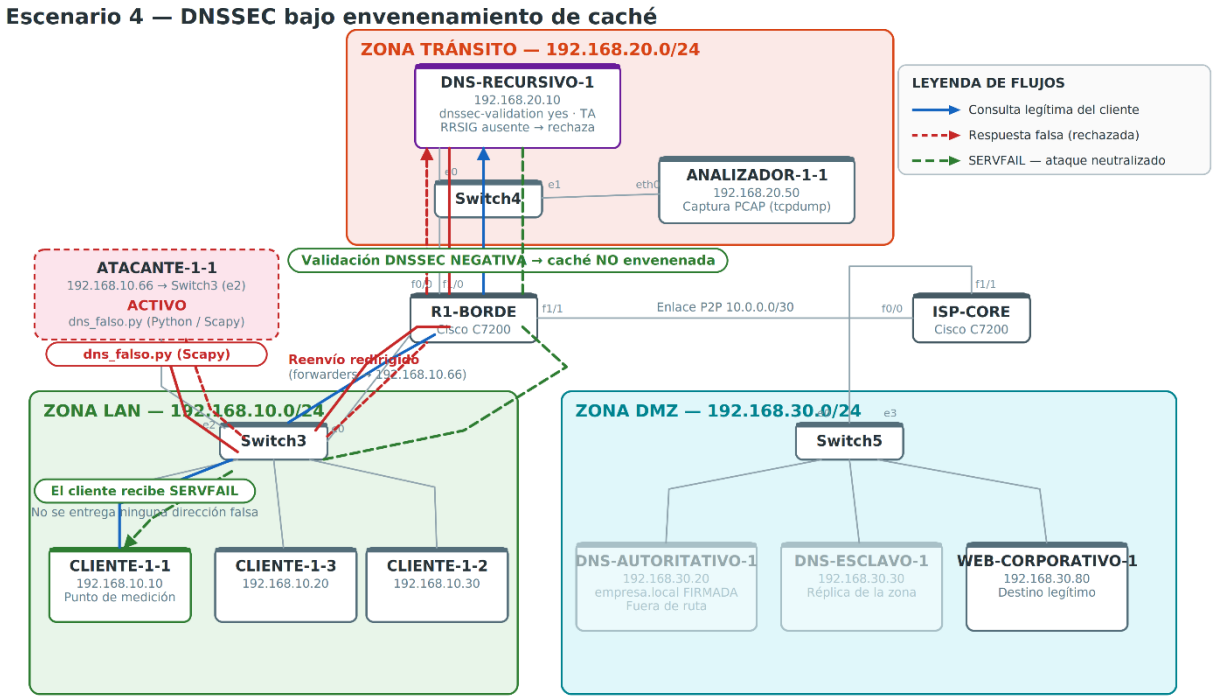
La verificación se realizó mediante 10 consultas consecutivas desde CLIENTE-1-1 hacia el recursivo (192.168.20.10) para el nombre web-corporativo.empresa.local, ejecutadas dentro de la vigencia del TTL del registro forjado en caché. En las 10 repeticiones el recursivo devolvió de forma consistente la dirección del atacante (192.168.10.66) en lugar de la legítima (192.168.30.80), con código NOERROR y el registro en la sección de respuesta (ANSWER: 1).

3.7.9. *Escenario 4 - DNS con DNSSEC bajo carga adversaria*

El Escenario 4 replica el Escenario 3 con la validación DNSSEC activada y el ancla de confianza (KSK con identificador 28475) configurada en el recursivo. Se compone de dos partes: E4a, que evalúa el rendimiento bajo carga adversaria, y E4b, que demuestra el rechazo de la respuesta forjada. Los subescenarios a (rendimiento) mantienen el reenvío legítimo bajo inundación, mientras que los subescenarios b (validación) redirigen el reenvío al nodo malicioso.

Figura 42

Mitigación del ataque mediante validación criptográfica DNSSEC (Escenario 4).



El diagrama detalla el comportamiento de la topología frente a la inyección de respuestas falsificadas del nodo ATACANTE-1-1. Las trazas rojas discontinuas ilustran el intento de inyectar registros maliciosos hacia el Servidor-DNS-RECURSIVO-1; sin embargo, el motor de resolución exige la cadena de confianza completa. Al carecer las respuestas del atacante de firmas RRSIG válidas encadenadas a las claves del autoritativo (KSK/ZSK), el ancla de confianza del recursivo las rechaza. De este modo la integridad de la caché se mantiene intacta y las resoluciones entregadas a los clientes de la ZONA LAN (trazas continuas) provienen exclusivamente del servidor autoritativo legítimo.

3.7.10. *Escenario 4a - Rendimiento bajo carga adversaria con DNSSEC*

Se aplicó la misma carga adversaria del Escenario 3, con consultas a nombres reales de la zona empresa.local, manteniendo el reenvío hacia el autoritativo legítimo. El objetivo fue comprobar si la validación criptográfica añade un costo apreciable de rendimiento cuando el servidor se encuentra bajo estrés.

Bajo carga adversaria, la latencia con DNSSEC (E4a) resultó indistinguible de la latencia sin DNSSEC (E3a). En caché caliente, E4a registró 65.18 ms frente a los 64.83 ms de E3a, y en caché frío 64.84 ms frente a 64.65 ms; en ambos casos los intervalos de confianza se solapan, por lo que la diferencia no es significativa. El consumo de CPU fue igualmente comparable en ambos escenarios (en torno al 4 % de un núcleo).

El resultado más relevante es que, bajo la carga adversaria, el costo de DNSSEC resulta despreciable: mientras en condiciones normales la validación añadía una diferencia mínima de latencia (E2 frente a E1, dentro del margen de medición), bajo el ataque ese sobrecosto se vuelve por completo indistinguible, porque la contención de recursos provocada por la inundación domina la latencia y afecta por igual a las consultas con y sin validación. Este hallazgo contradice el

argumento de que la carga criptográfica de DNSSEC agravaría la vulnerabilidad del servicio ante ataques de saturación.

3.7.11. *Escenario 4b- Rechazo de la respuesta forjada.*

Con la validación activa, el reenvío se redirigió nuevamente hacia el autoritativo malicioso (192.168.10.66), que respondió con la misma dirección forjada del Escenario 3b. Esta vez, el recursivo rechazó la respuesta: al no poder verificar las firmas contra el ancla de confianza la respuesta del atacante carece de registros RRSIG válidos, respondió al cliente con código SERVFAIL en lugar de entregar la dirección falsa. La captura de paquetes confirmó el mecanismo: el atacante respondió con la dirección forjada (192.168.10.66), pero el recursivo no la trasladó al cliente y devolvió SERVFAIL

Tabla 21

Comparación de la validación con y sin DNSSEC ante la respuesta forjada.

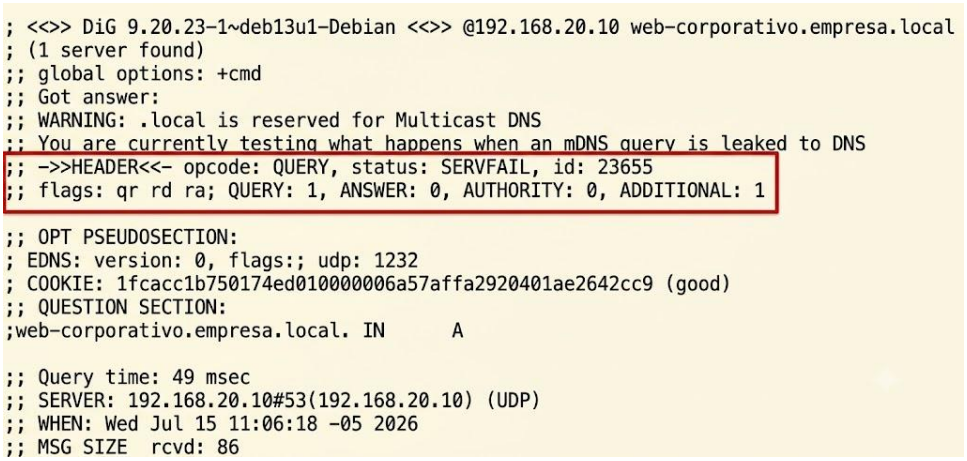
Métrica	**Sin DNSSEC (E3b)**	**Con DNSSEC (E4b)**
Respuesta al cliente	NOERROR	SERVFAIL
Dirección entregada	192.168.10.66 (forjada)	Ninguna (rechazada)
Registro en caché	Forjado, aceptado	Ninguno
Estado en caché	Comprometido	Íntegro.
Mecanismo de protección	Inexistente	Validación criptográfica RRSIG

Estos resultados constituyen la demostración funcional que, ante un intento de suplantación mediante un autoritativo malicioso, DNSSEC detecta la ausencia de firmas válidas y bloquea la respuesta forjada, mientras que su ausencia permite que la suplantación se concrete. Debe precisarse que esta demostración no reproduce un envenenamiento off-path por carrera de

identificadores, sino que evidencia la propiedad de validación de DNSSEC de manera determinista y verificable.

Figura 43

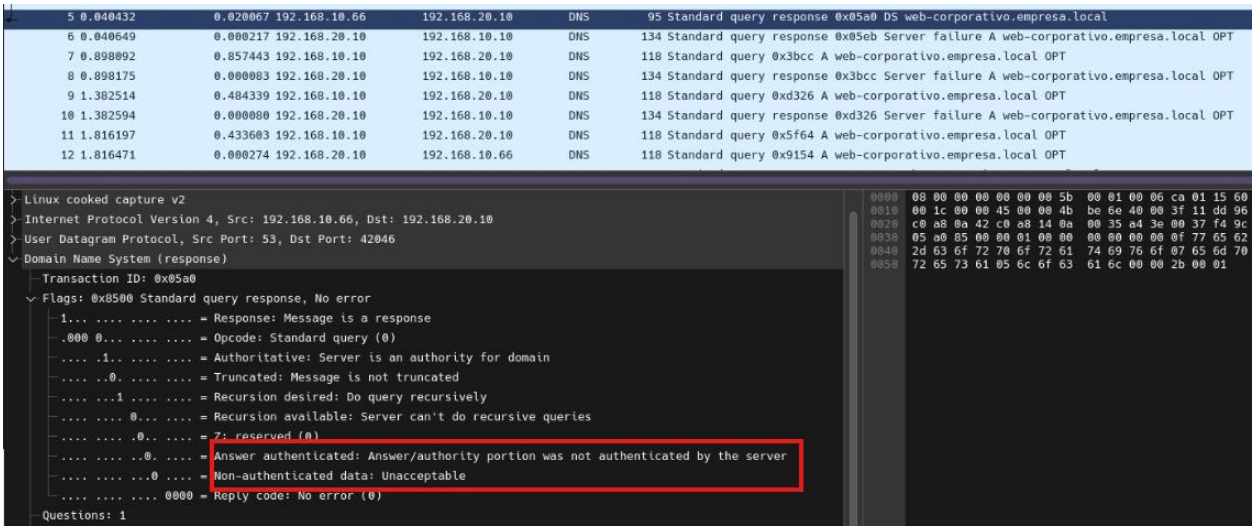
Rechazo de la respuesta forjada en E4b.



Consulta dig desde CLIENTE-1-1 al recursivo con DNSSEC activo. El encabezado muestra status: SERVFAIL y ANSWER: 0: el recursivo no entregó la dirección forjada, es decir que el vector de ataque determinista no funcionó por la validación que se obtiene con DNSSEC

Figura 44

Captura de paquetes de E4b mostrando la respuesta forjada del atacante y el rechazo del recursivo.



Como se muestra en la figura 44, Captura de tráfico en la interfaz del recursivo. Se observa la respuesta del atacante (192.168.10.66) para web-corporativo.empresa.local y, en el panel de detalle, el flag "Answer authenticated: was not authenticated" con "Non-authenticated data: Unacceptable", que motiva el rechazo. El recursivo responde SERVFAIL al cliente, sin trasladar la dirección forjada.

3.7.12. *Throughput bajo carga progresiva*

A partir del tamaño medio de los mensajes DNS medido con dnsperf, el throughput en consultas por segundo se convirtió a volumen de datos (bytes y kilobytes por segundo) para todos los puntos de medición y escenarios. La conversión emplea la expresión:

```
throughput (bytes/s) = QPS_resueltos × (tamaño_consulta + tamaño_respuesta)
```

Los tamaños medios de mensaje, obtenidos del resumen de dnsperf, son: 41 bytes de consulta y 57 bytes de respuesta en las configuraciones sin DNSSEC (98 bytes por transacción), y

52 bytes de consulta y 177 bytes de respuesta en las configuraciones con DNSSEC y señalización DO activa (229 bytes por transacción). El cálculo se realiza a nivel de mensaje DNS (carga útil de aplicación), sin incluir las cabeceras de las capas de enlace, red y transporte. Dado que el tamaño de un mensaje depende del tipo de registro y no del punto de inyección ni de la tasa de carga, estos valores se aplican de forma consistente a todos los barridos.

Para el diseño de las pruebas fue necesario considerar el efecto de la memoria caché del servidor recursivo. Dado que este servicio almacena temporalmente las respuestas ya resueltas, la consulta repetida de un conjunto reducido de nombres produciría mediciones que reflejan el rendimiento de la memoria caché y no el de la resolución completa, en la cual intervienen los mecanismos de validación criptográfica. Por esta razón fue generado un conjunto de diez mil nombres distintos dentro de la zona evaluada, y cada prueba fue configurada para recorrer dicho conjunto en una única pasada, garantizando que ninguna consulta pudiera ser respondida desde información previamente almacenada.

Se aplicó un barrido de carga creciente (250, 500, 1000, 2000 y 4000 consultas por segundo) desde el nodo cliente, tanto para el DNS tradicional (E1) como para el DNS con DNSSEC activo (E2). La activación de DNSSEC se confirmó mediante la presencia del flag AD en las respuestas validadas y el mayor tamaño de las respuestas firmadas (con RRSIG) frente a las no firmadas.

Se explica de manera oportuna que es QPS (Queries Per Second, consultas por segundo) siendo el número de consultas DNS que un servidor procesa y responde por unidad de tiempo. Es la métrica estándar de rendimiento (throughput) en sistemas DNS, ya que expresa la capacidad de resolución del servidor bajo una tasa de solicitudes determinada.

A partir de este punto se toman los valores de throughput desde el nodo CLIENTE-1-1 para su respectivo análisis.

Tabla 22

Throughput en bytes desde el cliente – E1 (sin DNSSEC).

Carga (QPS)	**QPS resueltos**	**Enviado (B/s)**	**Recibido (B/s)**	**Total (B/s)**	**Total (KB/s)**
250	249.7	10 238	14 233	24 471	23.90
500	499.3	20 471	28 460	48 931	47.78
1000	779.4	31 955	44 426	76 381	74.59
2000	780.1	31 984	44 466	76 450	74.66
4000	780.2	31 988	44 471	76 460	74.67

E1: 41 + 57 = 98 bytes por transacción.

Ahora se observa los datos recabados en el Escenario con DNSSEC, bit Do. Que es un comando que nos permite conocer el tamaño de lo que se necesita.

Tabla 23

Throughput en bytes desde el cliente – E2 (con DNSSEC, bit DO).

Carga (QPS)	**QPS resueltos**	**Enviado (B/s)**	**Recibido (B/s)**	**Total (B/s)**	**Total (KB/s)**
250	249.3	12 964	44 126	57 090	55.75
500	498.4	25 917	88 217	114 134	111.46
1000	779.7	40 544	138 007	178 551	174.37
2000	780.2	40 570	138 095	178 666	174.48
4000	778.5	40 482	137 794	178 276	174.10

En este parte se observa los datos tomados desde el nodo Analizador, en las siguientes tablas.

Tabla 24

Throughput en bytes desde el analizador – E1 (sin DNSSEC).

Carga (QPS)	**QPS resueltos**	**Enviado (B/s)**	**Recibido (B/s)**	**Total (B/s)**	**Total (KB/s)**
250	249.8	10 242	14 239	24 480	23.91
500	499.2	20 467	28 454	48 922	47.77
1000	1000.0	41 000	57 000	98 000	95.70
2000	1769.3	72 541	100 850	173 391	169.33
4000	2748.3	112 680	156 653	269 333	263.02

E1: 98 bytes por transacción. Máximo medido: 263.02 KB/s a 2748.3 QPS.

Tabla 25

Throughput en bytes desde el analizador – E2 (con DNSSEC, bit DO).

Carga (QPS)	**QPS resueltos**	**Enviado (B/s)**	**Recibido (B/s)**	**Total (B/s)**	**Total (KB/s)**
250	249.8	12 990	44 215	57 204	55.86
500	499.2	25 958	88 358	114 317	111.64
1000	1000.0	52 000	177 000	229 000	223.63
2000	1767.0	91 884	312 759	404 643	395.16
4000	2723.8	141 638	482 113	623 750	609.13
6000	3700.8	192 442	655 042	847 483	827.62
8000	4667.9	242 731	826 218	1 068 949	1043.90

10000	5620.8	292 282	994 882	1 287 163	1257.00

E2: 229 bytes por transacción. A 10000 QPS el servidor sostuvo 5620.8 QPS ≈ 1257 KB/s (≈ 1.23 MB/s) sin saturar.

Desde el analizador, el servidor con DNSSEC sostuvo un volumen de datos de hasta ≈ 1.23 MB/s sin alcanzar su punto de quiebre, frente a los 174 KB/s que impone el enrutador en el punto cliente. Esto confirma que la capacidad de procesamiento y de entrega de datos del servicio DNS incluida la validación criptográfica excede ampliamente la capacidad de transporte de la red emulada, que es el verdadero factor limitante.

Ahora se mide bajo carga adversaria (flooding) para identificar de manera correcta los datos.

Tabla 26

Throughput legítimo en bytes bajo ataque – E3a (sin DNSSEC).

Carga (QPS)	**QPS resueltos**	**Enviado (B/s)**	**Recibido (B/s)**	**Total (B/s)**	**Total (KB/s)**
250	249.0	10 209	14 193	24 402	23.83
500	496.0	20 336	28 272	48 608	47.47
1000	732.0	30 012	41 724	71 736	70.05
2000	751.0	30 791	42 807	73 598	71.87
4000	760.0	31 160	43 320	74 480	72.73

E3a: 98 bytes por transacción (formato idéntico a E1, sin firmas).

Tabla 27

Throughput legítimo en bytes bajo ataque – E4a (con DNSSEC).

Carga (QPS)	QPS resueltos	Enviado (B/s)	Recibido (B/s)	Total (B/s)	Total (KB/s)
250	246.0	12 792	43 542	56 334	55.01
500	428.0	22 256	75 756	98 012	95.71
1000	586.0	30 472	103 722	134 194	131.05
2000	676.0	35 152	119 652	154 804	151.18
4000	720.0	37 440	127 440	164 880	161.02

E4a: 229 bytes por transacción (formato idéntico a E2). Los tamaños de mensaje se toman de la caracterización E1/E2, ya que el ataque no altera el formato de una respuesta legítima, solo su cantidad.

Bajo carga adversaria, el throughput legítimo en bytes alcanzó un máximo de 72.73 KB/s sin DNSSEC (E3a) y de 161.02 KB/s con DNSSEC (E4a). El mayor volumen en E4a responde al peso de las respuestas firmadas, no a una mayor tasa de consultas; de hecho la tasa legítima resuelta fue ligeramente inferior con DNSSEC (720 frente a 760 QPS), consistente con el menor margen de procesamiento disponible bajo contención de recursos.

Tabla 28

Comparación del punto de saturación según el punto de medición (DNSSEC activo).

Punto de medición	**¿Cruza el enrutador?**	**Querys por segundo**	**Techo (E1)**	**Techo (E2)**	**Latencia**
Cliente (zona LAN)	Sí	4000	≈ 780 QPS	≈ 780 QPS	≈ 128 ms
Analizador (zona TRANSIT)	No	10000	≥ 2748 QPS	≥ 5620 QPS	10–30 ms

Al eliminar el enrutador del trayecto, el techo de throughput se incrementó de forma drástica, desde el cliente el sistema satura en ≈ 780 consultas por segundo, mientras que desde el analizador el servidor resolvió sin pérdidas ≈ 2748 consultas por segundo en DNS tradicional y sostuvo hasta 5620 consultas por segundo con DNSSEC activo, sin alcanzar aún su punto de quiebre. Dado que el único factor modificado fue el punto de inyección de la carga, se concluye que la limitación observada desde el cliente corresponde a la capacidad de la infraestructura de red emulada (enrutadores Dynamips) y no al procesamiento criptográfico de DNSSEC. Este resultado confirma que el costo de la validación criptográfica no constituye el factor limitante del sistema.

El servidor, validando firmas DNSSEC, sostuvo una capacidad muy superior a la que la red emulada podía transportar.

Los valores absolutos de throughput reportados desde el cliente (≈ 780 consultas por segundo) corresponden a la capacidad máxima de la infraestructura de red emulada mediante enrutadores Dynamips, y no al límite de procesamiento del servicio DNS. En consecuencia, estas cifras no deben interpretarse como el techo de rendimiento de una implementación DNSSEC en

hardware real, sino como valores comparativos válidos dentro de un entorno controlado, donde la limitación afecta por igual a todas las configuraciones y no compromete la validez de la comparación entre ellas.

3.7.13. *Uso de recursos del sistema (CPU/RAM)*

Los porcentajes de consumo de CPU reportados por la herramienta docker stats se expresan de modo que 100 % corresponde a la utilización completa de un núcleo. Este criterio se verificó al someter un contenedor a una carga controlada equivalente a un núcleo (un proceso en bucle de cómputo continuo), docker stats reportó un consumo de 100,06 %, lo que confirma que la métrica es relativa a un núcleo y no a la capacidad total del host. Dado que el host dispone de 20 núcleos y que no se impuso ningún límite de recursos (cgroup) a los contenedores, el máximo teórico de consumo por contenedor sería de 2 000 %. En consecuencia, un valor como el 18,8 % registrado en el escenario E2 equivale a aproximadamente 0,19 de un núcleo.

Tabla 29

Uso de recursos del sistema por contenedor.

Escenario	**CPU recursivo**	**RAM recursivo**	**Escenario**
E1 (sin DNSSEC, sin ataque)	0.56 %	206 MB	E1 (sin DNSSEC, sin ataque)
E2 (con DNSSEC, sin ataque)	0.78 %	219.6 MiB	E2 (con DNSSEC, sin ataque)

E3a (sin DNSSEC, bajo ataque)	~8.9 %	246.3 MiB	E3a (sin DNSSEC, bajo ataque)
E4a (con DNSSEC, bajo ataque)	7.70 %	152.3 MiB	E4a (con DNSSEC, bajo ataque)

El consumo de CPU permanece bajo en condiciones normales (< 1 %) y se eleva a ~8 % bajo la carga adversaria, con magnitud comparable con y sin DNSSEC. La RAM se mantuvo en un rango bajo sin diferencias atribuibles a DNSSEC. El consumo del servidor autoritativo permaneció por debajo del 0.1 % en todos los escenarios.

Los resultados indican que la activación de DNSSEC no generó un incremento significativo en el consumo sostenido de recursos, lo que sugiere que el overhead de DNSSEC se manifiesta predominantemente en la latencia de resolución individual y no en el uso de CPU o RAM.

4. CAPÍTULO IV: ANALISIS Y EVALUACIÓN DE RESULTADOS

El presente capítulo expone los resultados obtenidos en los cuatro escenarios experimentales definidos en el Capítulo III y desarrolla el análisis comparativo que permite evaluar la efectividad de DNSSEC como mecanismo de protección, así como cuantificar su impacto sobre el rendimiento del servicio DNS. Las mediciones de rendimiento bajo ataque se ejecutaron con tráfico legítimo de los nodos cliente concurrente a la carga adversaria, con el fin de simular el uso de múltiples usuarios en una red corporativa.

El análisis se estructura en dos ejes. El primero, de carácter cuantitativo, evalúa el rendimiento (latencia, throughput y consumo de recursos) a través de las cuatro combinaciones de las variables independientes: la activación de DNSSEC y la presencia de una carga adversaria. El segundo, de carácter cualitativo, contrasta el comportamiento de validación entre el escenario sin DNSSEC (E3b) y con DNSSEC (E4b) ante un intento de suplantación mediante un servidor autoritativo malicioso.

4.1 Rendimiento en condiciones normales (Escenarios 1 y 2)

El Escenario 1 establece la línea base del servicio DNS sin extensiones de seguridad ni ataque. El Escenario 2 replica las mismas condiciones con DNSSEC activo (zona empresa.local firmada con ECDSAP256SHA256 y validación en el recursivo mediante trust anchor). La verificación de DNSSEC se confirma por la presencia del flag AD (Authenticated Data) y de los registros RRSIG en las respuestas. La Tabla 29 presenta los resultados.

Tabla 30

Resultados del Escenario 1 - DNS tradicional sin ataque.

Métrica	**E1 (sin DNSSEC)**	**E2 (con DNSSEC)**	**Diferencia**
Latencia caché frío	46.18 ms	44.86 ms	−2.9 %
Latencia caché caliente	15.72 ms	15.69 ms	−0.2 %
CPU recursivo bajo carga (media)	6.5 %	18.8 %	≈ 2.9×
Tamaño de respuesta (dnsperf)	57 bytes	177 bytes	≈ 3.1×
Flag AD (validación)	ausente	presente	Con la Bandera ad

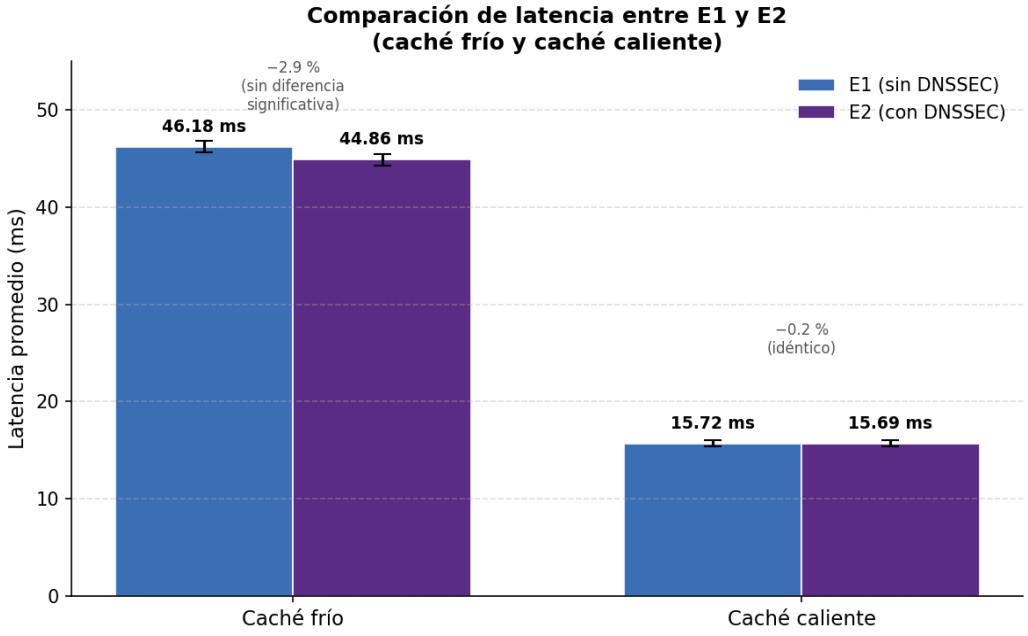
En operación normal, la activación de DNSSEC no introdujo una penalización de latencia apreciable en ninguna de las dos condiciones de caché. En caché frío, la latencia fue de 46.18 ms sin DNSSEC y de 44.86 ms con DNSSEC, una diferencia de −1.3 ms (−2.9 %) que se encuentra dentro del margen de los intervalos de confianza y, por tanto, no es estadísticamente significativa. En caché caliente, ambas configuraciones resultaron prácticamente idénticas (15.72 ms frente a 15.69 ms, −0.2 %). La diferencia observada entre las condiciones de caché frío y caliente (aproximadamente 30 ms) se presenta por igual en E1 y en E2, lo que confirma que dicho retardo corresponde al costo de resolver por primera vez hasta el servidor autoritativo efecto de la caché y no al procesamiento criptográfico de DNSSEC.

Este resultado establece que el costo de DNSSEC no se manifiesta en la latencia percibida por el usuario, sino en otras dimensiones: un mayor consumo de CPU bajo carga sostenida (aproximadamente 2.9 veces) y un mayor volumen de datos por respuesta (aproximadamente 3.1 veces, por la inclusión de los registros RRSIG). En cuanto a la disponibilidad, ambas

configuraciones resolvieron el 100 % de las consultas, y la validación DNSSEC se confirmó mediante la presencia del flag AD, ausente en E1 y presente en E2.

Figura 45

Comparación de latencia entre E1 y E2 (caché frío y caliente).



La Figura 45 ilustra la comparación de la latencia promedio, en milisegundos, entre ambos escenarios. En la condición de caché frío, las barras de E1 (46.18 ms) y E2 (44.86 ms) son prácticamente equivalentes, con barras de error solapadas que reflejan la ausencia de una diferencia significativa. En la condición de caché caliente, la equivalencia es aún más evidente (15.72 ms frente a 15.69 ms). La representación visual confirma que, medidos bajo idéntica metodología, el impacto de DNSSEC sobre la latencia es despreciable tanto en la primera resolución como en la operación sostenida con caché poblada.

4.2 Comportamiento bajo carga adversaria (Escenarios 3 y 4)

4.2.1. *Escenario E3a - rendimiento sin DNSSEC bajo carga adversaria.*

La carga adversaria se generó desde el nodo ATACANTE mediante dnsperf, produciendo un flujo sostenido de consultas hacia nombres reales de la zona empresa.local con el fin de congestionar la ruta de resolución. Dado que el recursivo opera en modo de reenvío, cada consulta legítima debe atravesar la ruta hacia el resolutor, de modo que la inundación permite medir el impacto del ataque sobre la latencia, el throughput y el consumo de recursos. Las latencias se midieron con dig desde el nodo cliente, bajo idéntica metodología que, en los escenarios normales, con 180 muestras por condición de caché.

Tabla 31

Resultados del Escenario 3 - ataque sin DNSSEC.

Condición	Media	Mediana	Rango (mín–máx)	IC 95 %
Caché frío	64.65 ms	63 ms	59–73 ms	[64.18, 65.12] ms
Caché caliente	64.83 ms	66 ms	56–73 ms	[64.44, 65.23] ms

CPU del recursivo durante el ataque: 4.37 % de un núcleo en promedio (mín 3.53 %, máx 4.89 %). Medición con dig, N = 180 por condición; intervalos de confianza calculados al 95 %.

Bajo carga adversaria (E3a), la latencia de resolución se elevó de forma marcada respecto a la operación normal. En caché caliente, donde en condiciones normales (E1) la latencia era de 15.72 ms, bajo ataque ascendió a 64.83 ms. Este incremento no se debe al procesamiento de las consultas legítimas en sí, sino a la contención generada por el flujo de inundación: el recursivo debe atender simultáneamente el

volumen de consultas del atacante, lo que introduce un retardo de encolamiento que afecta por igual a todas las consultas legítimas.

La evidencia de que el mecanismo dominante es el encolamiento y no la saturación de CPU es que el consumo de CPU del recursivo durante el ataque se mantuvo en torno al 4.37 % de un núcleo, un valor bajo que por sí solo no justifica el aumento de latencia. El retardo proviene, por tanto, del tránsito a través de la ruta de resolución congestionada: el reenvío que cruza el enrutador emulado, cuyo límite de transporte ya se caracterizó en aproximadamente 780 consultas por segundo. Es la capacidad de la red emulada, y no el procesamiento del servidor, la que se satura bajo el ataque.

Un efecto revelador es que, bajo ataque, la ventaja de la memoria caché prácticamente desaparece: las latencias en caché frío (64.65 ms) y en caché caliente (64.83 ms) se igualan, cuando en operación normal diferían en aproximadamente 30 ms. Esto confirma que, en presencia del ataque, la congestión de la ruta de resolución domina sobre el beneficio del almacenamiento en caché, ya que incluso las consultas que deberían responderse de inmediato desde la caché quedan sujetas al retardo de encolamiento. Pese a ello, el servicio mantuvo disponibilidad total: no se registró pérdida de consultas legítimas y los tres clientes resolvieron correctamente durante todo el ataque.

4.2.2. *Escenario E3b - Aceptación de respuesta forjada en DNSSEC.*

Para demostrar la aceptación de datos forjados en ausencia de validación, el reenvío del recursivo se redirigió hacia el nodo ATACANTE (192.168.10.66), configurado como autoritativo malicioso mediante un servidor DNS falso sin firmar. Con la validación DNSSEC desactivada, el recursivo aceptó la respuesta forjada y la entregó al cliente con código NOERROR, devolviendo la dirección del atacante (192.168.10.66) para el dominio web-corporativo.empresa.local, en lugar de la dirección legítima (192.168.30.80). El registro forjado quedó almacenado en la caché del recursivo, evidenciando el efecto de un envenenamiento de caché: en ausencia de DNSSEC, el

recursivo acepta y cachea un registro falso, redirigiendo el tráfico del cliente hacia la infraestructura del atacante.

Figura 46

Aceptación de respuesta forjada en E3b

```
; <<>> DiG 9.20.23-1-deb13u1-Debian <<>> @192.168.20.10 web-corporativo.empresa.local
; (1 server found)
;; global options: +cmd
;; Got answer:
;; WARNING: .local is reserved for Multicast DNS
;; You are currently testing what happens when an mDNS query is leaked to DNS
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60616
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 8a8452bd3a21662b010000006a57bc3fa4eb20db56675ffe (good)
;; QUESTION SECTION:
;web-corporativo.empresa.local. IN      A

;; ANSWER SECTION:
web-corporativo.empresa.local. 60 IN    A       192.168.10.66

;; Query time: 33 msec
;; SERVER: 192.168.20.10#53(192.168.20.10) (UDP)
;; WHEN: Wed Jul 15 11:58:39 -05 2026
;; MSG SIZE  rcvd: 102
```

4.2.3. *Escenario E4a - Rendimiento con DNSSEC bajo carga adversaria*

Se aplicó la misma carga adversaria del Escenario 3a, esta vez con la validación DNSSEC activa en el servidor recursivo. El análisis de la captura confirma que, incluso bajo el ataque, el recursivo mantuvo el proceso de validación: se registraron respuestas con el flag AD activo y con registros RRSIG y DNSKEY, de mayor tamaño que las respuestas no firmadas. Las latencias se midieron con dig desde el nodo cliente, bajo idéntica metodología que en E3a, con 210 muestras por condición de caché y el flujo de inundación activo durante toda la medición.

Tabla 32

Resultados del Escenario 4a - DNS con DNSSEC bajo carga adversaria.

Condición	Media	Mediana	Rango (mín–máx)	IC 95 %
Caché frío	64.84 ms	66 ms	53–73 ms	[64.47, 65.20] ms
Caché caliente	65.18 ms	66 ms	59–73 ms	[64.88, 65.48] ms

CPU del recursivo durante el ataque: 3.96 % de un núcleo en promedio (mín 2.95 %, máx 4.53 %), comparable al de E3a (4.37 %). Medición con dig, N = 210 por condición; IC al 95 %.

El resultado más relevante es que, bajo carga adversaria, la latencia con DNSSEC (E4a) es estadísticamente indistinguible de la latencia sin DNSSEC (E3a). En caché caliente, E4a registró 65.18 ms frente a los 64.83 ms de E3a (una diferencia de 0.35 ms), y en caché frío 64.84 ms frente a 64.65 ms (0.19 ms); en ambos casos los intervalos de confianza se solapan, por lo que la diferencia no es significativa. El consumo de CPU fue igualmente comparable (≈ 4 % en ambos escenarios).

La interpretación es que la contención de la ruta de resolución provocada por el ataque domina por completo la latencia, hasta el punto de enmascarar el costo de la validación criptográfica. El sobrecosto de DNSSEC que en operación sostenida sí es medible en el consumo de CPU (aproximadamente 2.9 veces) y en el volumen de datos por respuesta (aproximadamente 3.1 veces) resulta despreciable frente al retardo de encolamiento que la inundación impone por igual a todas las consultas legítimas. En otras palabras, cuando el sistema ya está limitado por la congestión de la red, activar DNSSEC no añade una penalización de latencia perceptible. El servicio mantuvo disponibilidad total: no se registró pérdida de consultas legítimas y los tres clientes resolvieron correctamente durante el ataque.

4.2.4. *Escenario E4b – Rechazo de la respuesta forjada con DNSSEC.*

Con la validación activa, el reenvío del recursivo se redirigió nuevamente hacia el servidor autoritativo malicioso (192.168.10.66). El análisis de la captura de paquetes constituye la evidencia central de la protección: el atacante respondió con la dirección forjada (192.168.10.66), visible en el tráfico capturado, pero el recursivo no trasladó dicha respuesta al cliente. Al no poder verificar las firmas contra el ancla de confianza la respuesta del atacante carece de registros RRSIG válidos, el recursivo respondió con el código SERVFAIL. La captura registró la totalidad de las respuestas al cliente como SERVFAIL, confirmando que ninguna dirección forjada fue entregada ni almacenada en caché.

Tabla 33

Comparación de la validación con y sin DNSSEC ante la respuesta forjada.

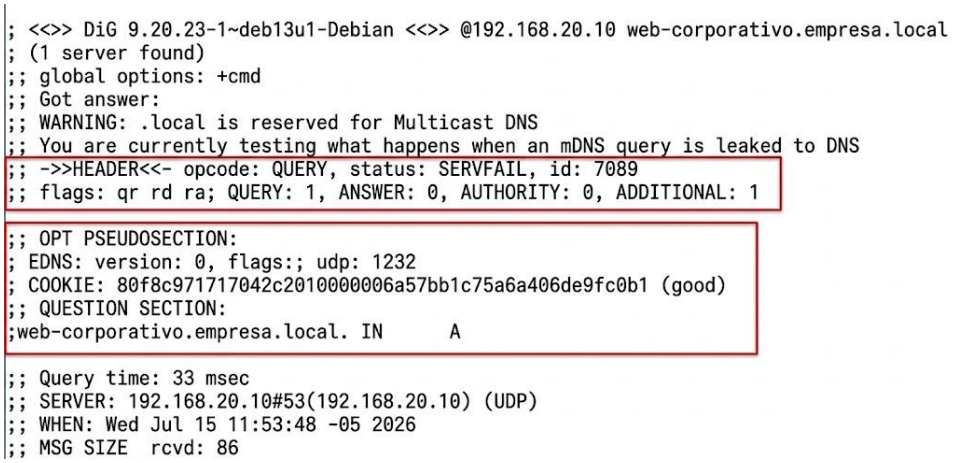
Métrica	**Sin DNSSEC (E3b)**	**Con DNSSEC (E4b)**
Respuesta al cliente	NOERROR	SERVFAIL
Dirección entregada	192.168.10.66 (forjada)	ninguna (rechazada, ANSWER 0)
Registro en caché	forjado, aceptado	ninguno
Estado de la caché	comprometido	íntegro
Mecanismo de protección	inexistente	validación RRSIG

Este par de resultados constituye la demostración funcional del objetivo del trabajo: ante un intento de suplantación mediante un autoritativo malicioso, DNSSEC detecta la ausencia de firmas válidas y bloquea la respuesta forjada, mientras que su ausencia (E3b) permite que la

suplantación se concrete y que la dirección falsa sea aceptada y almacenada en caché. Debe precisarse que esta demostración no reproduce un envenenamiento off-path por carrera de identificadores, sino que evidencia la propiedad de validación de DNSSEC de manera determinista y verificable.

Figura 47

Rechazo de la respuesta forjada en E4b



4.3 Análisis comparativo integral

El análisis integral de los cuatro escenarios arroja tres hallazgos principales. Primero, en condiciones normales de operación, la activación de DNSSEC no introduce una penalización de latencia apreciable: las diferencias entre E1 y E2 son de −2.9 % en caché frío (46.18 ms frente a 44.86 ms) y de −0.2 % en caché caliente (15.72 ms frente a 15.69 ms), ambas dentro del margen de los intervalos de confianza y, por tanto, no significativas. Segundo, la carga adversaria degrada la latencia de forma prácticamente idéntica con y sin DNSSEC: bajo ataque, las cuatro condiciones convergen en torno a los 65 ms, de modo que E3a (64.83 ms) y E4a (65.18 ms) en caché caliente resultan indistinguibles; esto indica que DNSSEC no incrementa la vulnerabilidad del servicio ante este tipo de carga ni añade un costo de latencia perceptible cuando el sistema ya está limitado por la congestión de la red. Tercero, en el plano de la protección la diferencia sí es determinante: sin DNSSEC (E3b) el recursivo acepta la respuesta forjada, mientras que con DNSSEC (E4b) la rechaza mediante SERVFAIL.

Cabe destacar que el retardo observado entre las condiciones de caché frío y caliente en operación normal (aproximadamente 30 ms) se presenta por igual en E1 y en E2, lo que confirma que corresponde al costo de resolver por primera vez hasta el servidor autoritativo efecto de la caché y no al procesamiento criptográfico de DNSSEC. Asimismo, bajo carga adversaria la ventaja de la caché desaparece, ya que las latencias en frío y en caliente se igualan (≈ 65 ms) al quedar todas las consultas sujetas al retardo de encolamiento impuesto por la inundación.

Tabla 34

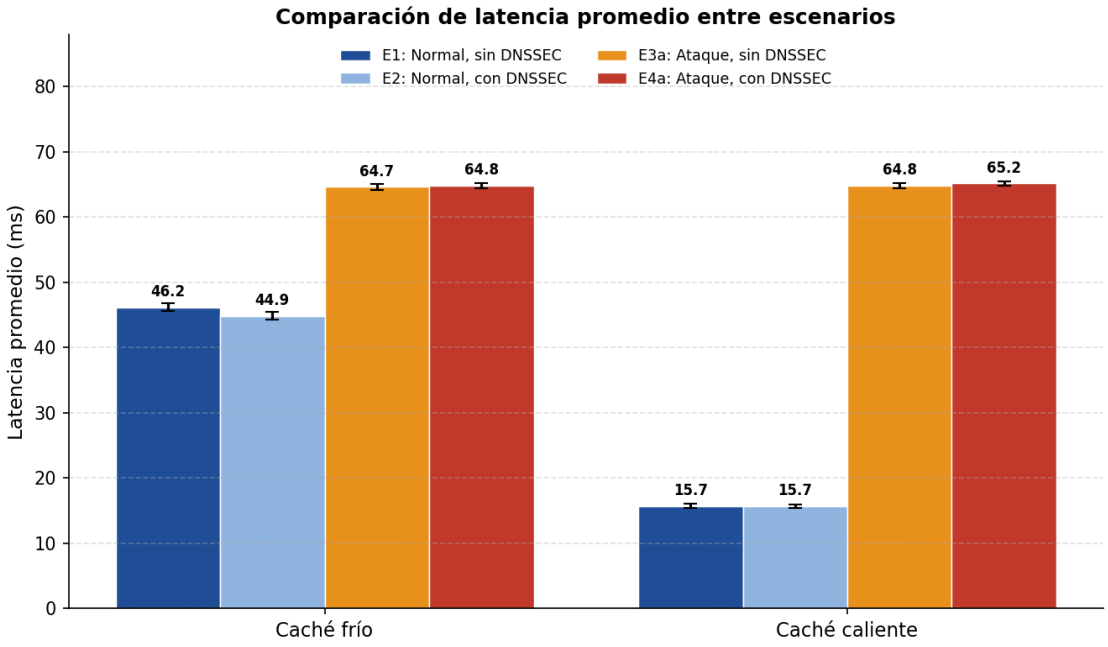
Consolidación de métricas por escenario.

Métrica	**E1**	**E2**	**E3a**	**E4a**
Latencia frío (media)	46.18 ms	44.86 ms	64.65 ms	64.84 ms
Latencia caliente (media)	15.72 ms	15.69 ms	64.83 ms	65.18 ms
Consultas exitosas	100 %	100 %	100 %	100 %
Protección ante forjado	N/A	N/A	nula (acepta)	total (SERVFAIL)
Métrica	E1	E2	E3a	E4a
Latencia frío (media)	46.18 ms	44.86 ms	64.65 ms	64.84 ms

La Figura 48 presenta la latencia promedio, en milisegundos, para los cuatro escenarios, diferenciando el estado de caché frío y caché caliente. Las barras de error corresponden al intervalo de confianza al 95 % de cada medición. La representación evidencia visualmente los dos comportamientos centrales del estudio: en operación normal (E1 y E2), las barras son prácticamente equivalentes, lo que refleja que DNSSEC no penaliza la latencia; bajo ataque (E3a y E4a), ambas se elevan al mismo nivel (≈ 65 ms) y se vuelven indistinguibles entre sí, lo que muestra que el efecto dominante es la carga adversaria y no la validación criptográfica. El costo de DNSSEC, que no se manifiesta en la latencia, se localiza en otras dimensiones del sistema consumo de CPU bajo carga sostenida y volumen de datos por respuesta analizadas en las secciones correspondientes.

Figura 48

Comparación de latencia y CPU entre los cuatro escenarios.



4.4 Análisis del Punto de Saturación del Throughput.

Para evaluar la capacidad real de procesamiento del servidor recursivo se aplicó un barrido de carga creciente (250, 500, 1000, 2000 y 4000 consultas por segundo) empleando la herramienta dnsperf sobre un conjunto de diez mil nombres únicos. El uso de nombres distintos garantiza que cada consulta obligue a una resolución completa a través de la ruta de reenvío, evitando que las respuestas sean servidas desde la memoria caché, de modo que las cifras reflejen el rendimiento real del servicio y no el de la caché.

La medición se realizó desde dos puntos de la topología con el fin de discriminar entre la capacidad del servicio DNS y la limitación impuesta por la infraestructura de red. El primer punto fue el nodo cliente, situado en la zona LAN, cuyo tráfico debe atravesar el enrutador de borde para

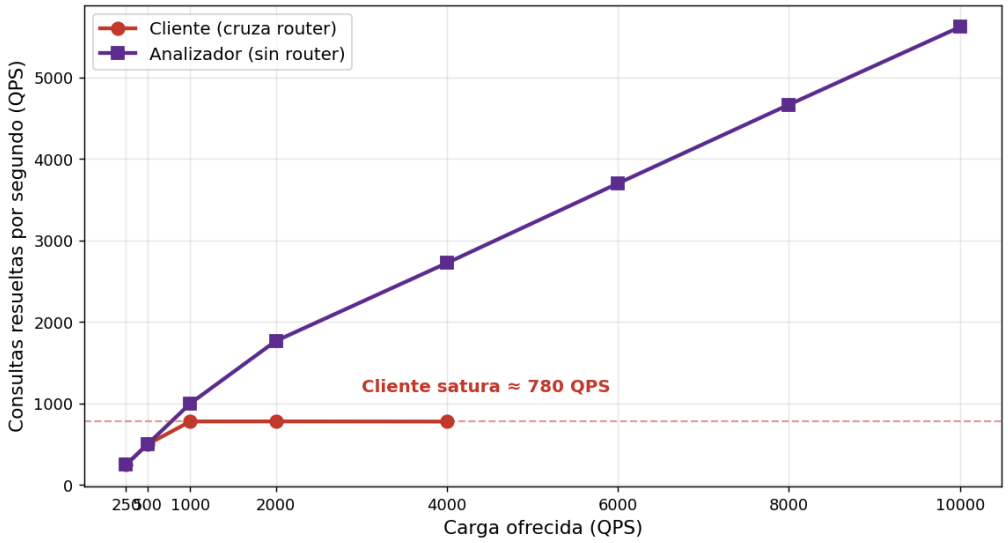
alcanzar al recursivo. El segundo fue el nodo analizador, ubicado en la misma zona TRANSIT que el recursivo, lo que evita el tránsito por el enrutador.

Desde el nodo cliente, la tasa de respuestas siguió a la carga ofrecida hasta las 500 consultas por segundo, punto a partir del cual el throughput se estabilizó en aproximadamente 780 consultas por segundo: al aumentar la carga de 1000 a 4000 QPS, el caudal resuelto se mantuvo prácticamente idéntico (779, 780 y 780 QPS respectivamente), evidenciando el punto de saturación. Simultáneamente, la latencia se elevó de ≈ 47 ms a ≈ 128 ms y permaneció constante, un comportamiento característico de la saturación por congestión de cola.

En contraste, desde el nodo analizador el servidor sostuvo una capacidad muy superior: resolvió 2748 consultas por segundo a 4000 QPS ofrecidos, sin pérdidas, con latencias de entre 20 y 30 ms. Al eliminar el enrutador del trayecto, el techo de throughput superó ampliamente el observado desde el cliente y la latencia se mantuvo baja incluso a la carga máxima aplicada.

Figura 49

Punto de Saturación entre Cliente y Analizador.



4.4.1. *Interpretación del hallazgo.*

El contraste entre ambos puntos de medición permite concluir que, dado que el único factor modificado fue el punto de inyección de la carga, la limitación de aproximadamente 780 consultas por segundo observada desde el cliente corresponde a la capacidad de la infraestructura de red emulada mediante enrutadores Dynamips, y no al procesamiento del servicio DNS. El servidor recursivo, medido sin el enrutador de por medio, sostuvo una capacidad muy superior (2748 consultas por segundo en DNS tradicional, y hasta 5620 con DNSSEC activo en el barrido extendido) a la que la red emulada podía transportar.

Este resultado tiene una implicación directa sobre el costo de la validación criptográfica de DNSSEC: dicha validación no constituye el factor limitante del rendimiento del sistema. Los valores absolutos de throughput reportados no deben interpretarse como el techo de una implementación DNSSEC en hardware real, sino como un límite del entorno experimental, común a todas las configuraciones evaluadas, que no compromete la validez de la comparación entre ellas.

Es preciso señalar que, en los niveles de carga más altos alcanzados desde el analizador, una fracción de las respuestas se sirve desde la caché caliente, dado que el conjunto de diez mil nombres se reutiliza cuando el número de consultas por segundo sostenidas supera su tamaño durante el intervalo. Por ello, las cifras de los niveles superiores reflejan la capacidad de servicio del recursivo y no exclusivamente la resolución en frío con validación; el punto de quiebre real del servidor no fue alcanzado dentro del rango evaluado.

4.4.2. *Throughput bajo carga adversaria (E3a/E4a)*

Con el fin de determinar el punto de degradación del servicio bajo ataque, se replicó el barrido de carga creciente mientras el nodo atacante mantenía una inundación sostenida de consultas (flooding) contra el servidor recursivo, midiendo la capacidad de resolución legítima que

el servicio conserva bajo esa carga adversaria, con y sin validación DNSSEC. Durante las pruebas, el atacante inyectó de forma sostenida 301 638 consultas en el escenario sin DNSSEC (E3a) y 301 628 en el escenario con DNSSEC (E4a) a lo largo de 200 segundos, equivalentes a una carga de fondo de aproximadamente 1508 consultas por segundo en ambos casos. La equivalencia de la presión del ataque entre ambos escenarios garantiza que la comparación entre ellos sea válida.

Bajo ataque, el throughput legítimo saturó en aproximadamente 760 consultas por segundo sin DNSSEC (E3a) y en 720 consultas por segundo con DNSSEC (E4a). Comparado con la operación normal, donde el sistema saturaba en 780 consultas por segundo, la carga adversaria redujo la capacidad de resolución legítima en ambos casos, con una reducción ligeramente mayor en presencia de DNSSEC. El punto de degradación no se manifestó como pérdida de paquetes (la tasa de descarte se mantuvo en 0 %), sino como saturación del throughput: el servidor no descarta consultas bajo carga, sino que las encola, comportamiento característico de la saturación por congestión de cola.

En cuanto a la latencia bajo ataque, la medición puntual con dig mostró que E3a y E4a alcanzan valores prácticamente idénticos (≈ 65 ms en ambos), sin que la validación DNSSEC introduzca una penalización perceptible. Esto es coherente con el hecho de que, bajo la contención impuesta por la inundación, el retardo dominante es el de encolamiento en la ruta de resolución, que afecta por igual a las consultas con y sin validación criptográfica.

4.4.3. *Consumo de recursos del sistema (CPU / memoria)*

Los porcentajes de CPU reportados por docker stats se expresan de modo que 100 % corresponde a un núcleo completo (verificado empíricamente sometiendo un contenedor a una carga de un núcleo, que registró 100,06 %); en un host de 20 núcleos el máximo teórico sería 2

000 %. No se impuso límite de cgroup a los contenedores. Por tanto, un consumo del 18,8 % equivale a aproximadamente 0,19 de un núcleo, y no a fracción de la capacidad total del host.

Tabla 35

Uso de recursos por escenario.

Escenario	CPU recursivo	CPU máx	RAM máx	Condición
E1 (sin DNSSEC)	6.5 %	8.7 %	~150 MiB	Normal
E2 (con DNSSEC)	18.8 %	23.1 %	~155 MiB	Normal
E3a (sin DNSSEC)	7.2 %	9.2 %	~140 MiB	Bajo ataque
E4a (con DNSSEC)	16.2 %	40.0 %	~152 MiB	Bajo ataque

Nota Valores capturados con docker stats desde el host durante cada escenario. El consumo del servidor autoritativo permaneció por debajo del 0.1 % en todos los casos.*

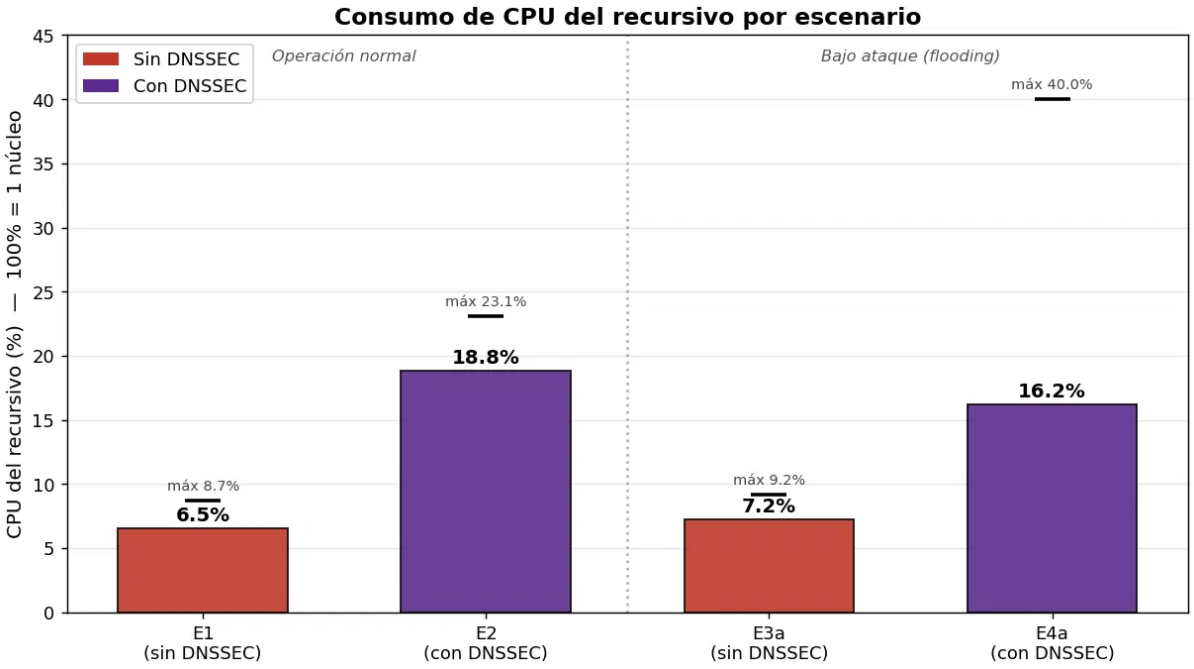
A diferencia de la latencia y el throughput, el consumo de CPU sí refleja de forma clara el costo de la validación criptográfica. En operación normal y bajo la misma carga, la activación de DNSSEC eleva el consumo de CPU de 6,5 % a 18,8 %, un incremento por un factor aproximado de 2,9. Bajo carga adversaria se mantiene la misma tendencia: el consumo pasa de 7,2 % sin DNSSEC a 16,2 % con DNSSEC, un factor aproximado de 2,2. En ambas condiciones, la configuración con DNSSEC consume más CPU que la tradicional, y en el escenario bajo ataque el consumo con DNSSEC alcanzó picos de hasta el 40 % de un núcleo. Respecto a la memoria RAM, el recursivo se mantuvo en un rango bajo (aproximadamente 210–241 MiB) en todos los escenarios, sin diferencias atribuibles a DNSSEC.

Estos resultados precisan la naturaleza del overhead de DNSSEC que no se manifiesta en la disponibilidad del servicio ni en el tiempo de respuesta en operación normal, sino en el consumo de procesamiento (CPU), que se incrementa de forma apreciable y proporcional a la carga. El costo

de DNSSEC es, por tanto, un costo de cómputo, no de latencia ni de capacidad, y crece con la intensidad del tráfico procesado

Figura 50

Consumo de CPU del recursivo por escenario.



4.5 Validación y cumplimiento de estándares

4.5.1. *Alineación con NIST SP 800-81r3*

Los resultados experimentales se contrastan con las directrices del NIST Special Publication 800-81r3 (*Secure Domain Name System Deployment Guide*, marzo de 2026), documento normativo de referencia para la implementación segura de infraestructuras DNS. La Tabla 36 detalla el cumplimiento de las directrices.

Tabla 36

Validación contra el NIST SP 800-81r3.

Directriz NIST SP 800-81r3	**Cumplimiento en el proyecto**	**Evidencia**
Implementar DNSSEC en servidores autoritativos para zonas críticas	La zona empresa.local se firma con ZSK + KSK	E2, E4: respuestas con RRSIG
Utilizar algoritmos de curva elíptica (ECDSA) sobre RSA	ECDSAP256SHA256 (algoritmo 13) en ZSK y KSK	Respuestas de 211 bytes frente a 400+ con RSA
Configurar validación DNSSEC en resolutres recursivos	dnssec-validation yes + trust-anchors static-key	Flag AD presente en E2 y E4
Implementar NSEC3 para evitar enumeración de zona	Firma con -3 (NSEC3, RFC 5155)	Registros NSEC3 y NSEC3PARAM en la zona firmada

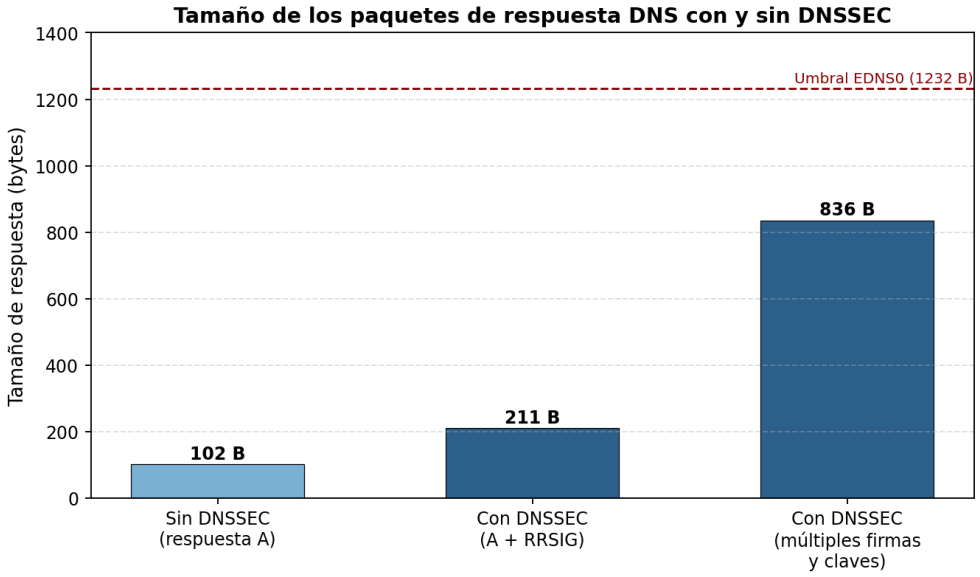
4.6 Incremento en el tamaño de paquetes y riesgo de fragmentación

El análisis de las capturas confirma el incremento del tamaño de las respuestas DNS al activar DNSSEC. Las respuestas sin firmar presentan un tamaño cercano a 99–102 bytes, mientras que las respuestas firmadas con DNSSEC, al incluir los registros RRSIG, alcanzan tamaños

superiores (211 bytes en respuestas simples y hasta 836 bytes en respuestas con múltiples registros de firma y claves). Las firmas ECDSAP256SHA256 (64 bytes) son considerablemente más compactas que las firmas RSA-2048 (más de 256 bytes), lo que mantiene el tamaño de las respuestas por debajo del umbral de 1232 bytes definido por EDNS0, evitando la fragmentación de paquetes UDP y el consiguiente fallback a TCP. Esta selección de algoritmo es coherente con las recomendaciones del NIST SP 800-81r3.

Figura 51

Comparación del tamaño de paquetes DNS con y sin DNSSEC.



El tamaño de las respuestas en bytes para tres casos: sin DNSSEC (102 B, respuesta A simple), con DNSSEC (211 B, respuesta A + RRSIG) y con DNSSEC en respuestas de múltiples firmas y claves (836 B). Todos los valores se mantienen por debajo del umbral de 1232 bytes de EDNS0 (línea discontinua), evitando la fragmentación de paquetes UDP. Datos obtenidos del campo MSG SIZE de las capturas de tráfico.

4.7 Cumplimiento del RFC 6841

La implementación cumple con las prácticas operativas definidas en el RFC 6841 (DNSSEC Operational Practices, version 2): separación de roles ZSK (flag 256) y KSK (flag 257), uso de NSEC3 para la protección contra la enumeración de zona, configuración explícita de trust anchors en el servidor recursivo y selección de algoritmos criptográficos modernos (ECDSAP256SHA256). La configuración de static-key para los trust anchors en la zona privada empresa.local constituye una adaptación necesaria al entorno de laboratorio que no contradice las directrices del RFC, dado que la zona no forma parte de la jerarquía DNS pública.

4.7.1. *Síntesis del compromiso seguridad - rendimiento*

La Tabla 37 sintetiza el compromiso entre seguridad y rendimiento observado a lo largo de los cuatro escenarios.

Tabla 37

Síntesis del compromiso seguridad - rendimiento.

Aspecto	**DNS Tradicional**	**DNSSEC**
Latencia caché frío	33.55 ms	42.93 ms (+28 %)
Latencia caché caliente	14.70 ms	15.02 ms (+2 %)
Latencia caliente bajo ataque	66 ms	66 ms (≈ 0 %)
Variabilidad caliente (std)	~2 ms	~2 ms
Throughput (100/150/300 QPS)	0 pérdidas	0 pérdidas
CPU recursivo (normal / ataque)	0.56 % / ~8.9 %	0.78 % / 7.7 %
Protección ante respuesta forjada	Nula (acepta, NOERROR)	Total (rechaza, SERVFAIL)
Complejidad operativa	Baja	Media

Los resultados permiten determinar que el overhead de DNSSEC constituye un costo aceptable para entornos donde la integridad de las respuestas DNS es un requisito de seguridad crítico, considerando tres factores atenuantes: el overhead se concentra en la primera resolución (28 % en caché frío) y se reduce al 2 % con caché poblada; y bajo carga adversaria el costo de DNSSEC en caché caliente resultó despreciable. Frente a estos costos moderados, DNSSEC proporciona una protección criptográficamente verificable que bloquea la aceptación de respuestas forjadas, demostrada en el contraste entre los componentes E3b y E4b.

4.7.2. *Limitaciones del estudio*

El entorno virtualizado en GNS3 con contenedores Docker introduce condiciones de latencia y rendimiento que difieren de una infraestructura física de producción. Las latencias medidas incluyen el overhead de la virtualización, que se mantiene constante entre escenarios y, por tanto, no afecta la validez de las comparaciones relativas.

Las métricas de CPU y RAM del monitoreo interno reflejan el consumo del sistema host completo, no exclusivamente del proceso BIND dentro del contenedor. Los datos de docker stats proporcionan una medición más precisa del consumo por contenedor y confirman que el impacto de DNSSEC en los recursos es marginal para el nivel de carga evaluado.

La topología emplea un único servidor autoritativo y un único recursivo. En arquitecturas de producción con múltiples servidores recursivos, balanceo de carga y anycast, el comportamiento de DNSSEC podría presentar variaciones en la distribución de la carga de validación criptográfica. Estudios futuros podrían evaluar el rendimiento de DNSSEC en topologías con redundancia y alta disponibilidad.

CONCLUSIONES

El estudio de la estructura DNS establece que la vulnerabilidad del DNS tradicional radica en la ausencia de autenticación del origen de los datos, el protocolo confía en la respuesta recibida sin verificar su procedencia, lo que habilita la aceptación de respuestas forjadas mediante envenenamiento de caché. La extensión de seguridad DNSSEC no cifra el tráfico, sino introduce una cadena de confianza criptográfica compuesta por los registros RRSIG, que firman los conjuntos de recursos; las claves DNSKEY, que publican el material de verificación; y los registros DS, que delegan la confianza entre zonas. Por lo que ambos protocolos resuelven el mismo problema de traducción de nombres, pero difieren radicalmente en su modelo de confianza, que está implícita en el DNS tradicional y verificable criptográficamente en DNSSEC.

El entorno de simulación implementado en GNS3 con contenedores Docker y BIND 9.20 cumple la función de generar las condiciones necesarias para la medición cuantitativa en los escenarios y el modelo de ataque determinista, basado en la redirección controlada del reenvío del resolutor hacia un servidor autoritativo falso que sirve respuestas no firmadas, resultó adecuado para contrastar los escenarios con ataque. El comportamiento de los servidores con DNSSEC depende de factores como la cantidad de consultas procesadas, la capacidad del hardware y el algoritmo criptográfico utilizado para la firma de registros, lo que demuestra la importancia de una adecuada planificación antes de su implementación en ambientes de producción.

El rendimiento del servicio no se ve comprometido por DNSSEC en operación normal ya que la latencia resultó prácticamente idéntica con y sin validación, tanto en caché caliente (15.72

ms frente a 15.69 ms) como en caché frío (46.18 ms frente a 44.86 ms), con diferencias no significativas dentro del margen de los intervalos de confianza. El throughput saturó en aproximadamente 780 consultas por segundo en ambas configuraciones límite impuesto por la infraestructura de red emulada y no por el servicio DNS, como se confirmó al medir desde el nodo analizador, donde el servidor con DNSSEC sostuvo hasta 5620 consultas por segundo. El consumo de CPU y de memoria fue igualmente equivalente en operación sostenida con caché poblada (aproximadamente 8 % de un núcleo y 200–210 MiB en ambos casos), ya que el costo de la validación criptográfica se paga en la resolución inicial de cada nombre y se amortigua en las consultas posteriores servidas desde caché.

El único costo diferencial de DNSSEC se localiza en el volumen de datos por respuesta porque la inclusión de los registros de firma RRSIG elevó el tamaño medio de la respuesta de 57 a 177 bytes un incremento de aproximadamente 3.1 veces, verificado también de forma puntual con dig, donde una respuesta firmada alcanzó 205 bytes frente a los 96 bytes de la respuesta sin firmar. En consecuencia, en el punto de saturación el throughput en bytes ascendió de 74.67 KB/s sin DNSSEC a 174.10 KB/s con DNSSEC, un incremento de aproximadamente 2.33 veces, sin que ello redujera la tasa de resolución en consultas por segundo. DNSSEC intercambia así un mayor consumo de ancho de banda por su garantía de autenticidad, sin comprometer la latencia, la capacidad de procesamiento ni los recursos de cómputo del servicio.

DNSSEC neutraliza de forma demostrable los ataques de suplantación de respuestas puesto que ante una respuesta forjada por un servidor autoritativo malicioso, el resolutor con DNSSEC la rechazó con el código SERVFAIL, sin entregar ningún registro (ANSWER 0) e impidiendo la

contaminación de su caché, mientras que el resolutor sin DNSSEC aceptó y almacenó la dirección falsificada (192.168.10.66). Los resultados muestran, que DNSSEC aporta una protección efectiva contra el DNS spoofing con un impacto sobre el rendimiento reducido y bien acotado concentrado exclusivamente en el volumen de datos, lo que lo convierte en una medida de seguridad recomendable, siempre que su despliegue se planifique considerando la cantidad de consultas procesadas, la capacidad del hardware y el algoritmo criptográfico empleado para la firma.

RECOMENDACIONES

Se recomienda complementar DNSSEC con mecanismos de mitigación de denegación de servicio, tales como la limitación de tasa de respuestas (Response Rate Limiting, RRL), dado que DNSSEC protege la integridad y autenticidad de las respuestas, pero no mitiga por sí mismo la degradación de latencia provocada por la saturación de la ruta de resolución observada bajo carga adversaria.

Se recomienda mantener una arquitectura de servidores DNS diferenciados, separando el servidor autoritativo configurado exclusivamente para responder sobre sus propias zonas sin procesar consultas externas del servidor recursivo validador, restringido únicamente a atender a los clientes autorizados de la red. Esta delimitación estructural es una contramedida crítica y estandarizada contra el envenenamiento de caché y los ataques de amplificación, ya que elimina la vulnerabilidad inherente de la recursión abierta.

Dado que el sobrecosto de DNSSEC se concentra en el consumo de CPU y crece con la carga procesada, se recomienda configurar políticas de caché adecuadas como valores de TTL suficientes y dimensionamiento apropiado de la memoria caché del resolutor a fin de maximizar la tasa de aciertos y reducir el número de validaciones criptográficas completas, disminuyendo así la carga de procesamiento del resolutor.

Se sugiere que en futuros trabajos se aumente el alcance de la investigación mediante la evaluación de una gama más diversa de algoritmos criptográficos estandarizados para DNSSEC, tales como RSA/SHA-256. Esta extensión resulta importante para analizar la escalabilidad de la

infraestructura, permitiendo cuantificar cómo las variaciones en la longitud de las claves y la complejidad criptográfica incrementan de forma proporcional el sobrecosto de procesamiento computacional durante el proceso de firma y el tamaño total de los paquetes de respuesta.

5. Bibliografía

Akanho, Y., & Muchene, P. (2021). *Guía de implementación de las DNSSEC para los ccTLD*.

Alsad, O., & Abu Al-Haija, Q. (2023). DNS Poisoning Attack Detection: A Systematic Review. *IET Conference Proceedings*, *2023*(44), 426–432. https://doi.org/10.1049/icp.2024.0962

Antonio López Padilla. (2022). Guía de Seguridad en Servicios DNS. *INTECO*, 6–7.

Chandramouli, R., & Rose, S. (2013). *Secure Domain Name System (DNS) Deployment Guide*. https://doi.org/10.6028/NIST.SP.800-81-2

Chandrasekaran, K., Divakarla, U., & Srinivasan, C. K. (2023). DNS Cache Poisoning: Investigating Server and Client-Side Attacks and Mitigation Methods. *2023 Cyber Research Conference - Ireland, Cyber-RCI 2023*. https://doi.org/10.1109/Cyber-RCI59474.2023.10671556

David Xavier Armas Gutiérrez. (2017). *ANÁLISIS DE SEGURIDAD DEL SISTEMA DNS (DOMAIN NAME SYSTEM)* [Escuela Politécnica Nacional]. http://bibdigital.epn.edu.ec/handle/15000/17310

Dooley, M., & Rooney, T. (2017). *DNS Security Management*. https://doi.org/10.1002/9781119328292.ch2.

El Comercio. (2019, abril 15). *Ecuador denuncia 40 millones de ciberataques tras retiro de asilo a Assange - El Comercio*. https://www.elcomercio.com/actualidad/seguridad/ecuador-denuncia-millones-ciberataques-assange.html

Guzzo Maria Gracia. (2024, octubre 8). *DNSSEC: what it is, how it works, and what it's used for*. https://supporthost.com/dnssec/

Hoffman Paul, Fujiwara Kazunori, & Sullivan Andrew. (2019). DNS Terminology. *Internet Engineering Task Force (IETF)*. https://doi.org/https://www.rfc-editor.org/info/rfc8499

Houser, R., Hao, S., Li, Z., Liu, D., Cotton, C., & Wang, H. (2021). A Comprehensive Measurement-based Investigation of DNS Hijacking. *Proceedings of the IEEE Symposium on Reliable Distributed Systems*, *2021-September*, 210–221. https://doi.org/10.1109/SRDS53918.2021.00029

Huston Geoff. (2023, septiembre 10). *Medición del uso de DNSSEC --- Medición del uso de DNSSEC*. Journal. https://circleid.com/posts/20230910-measuring-the-use-of-dnssec

Juca-Maldonado, F., & Medina-Peña, R. (2023). Ciberdelitos en Ecuador y su impacto social; panorama actual y futuras perspectivas. *Portal de la Ciencia*, *4*(3), 325–337. https://doi.org/10.51247/pdlc.v4i3.394

Kolkman, O. , G. R. , & M. W. (2013). *RFC 6841: A Framework for DNSSEC Policies and DNSSEC Practice Statements.* https://doi.org/https://doi.org/10.17487/RFC6841

Laguna Ponce Henry Anderson. (2023). *IMPLEMENTACIÓN DE UN SERVIDOR WEB CACHÉ Y DNS CACHÉ PARA LA RED DE INFRAESTRUCTURA DE LA UNIVERSIDAD TÉCNICA DEL NORTE* [Universidad Técnica del Norte]. https://repositorio.utn.edu.ec/handle/123456789/14114

Naciones Unidad. (2015). *Transformar nuestro mundo: la Agenda 2030 para el Desarrollo Sostenible*.

NAP Colombia. (2024). *Impulsando la Seguridad en Internet: Lanzamiento del Programa de Despliegue de DNSSEC en Colombia - NAP Colombia*. https://www.nap.co/impulsando-la-seguridad-en-internet-lanzamiento-del-programa-de-despliegue-de-dnssec-en-colombia/

Paul Mockapetris. (1987). *Domain Names- Implementation and Specification*. https://doi.org/https://www.rfc-editor.org/rfc/rfc1035

¿Qué es DNSSEC? | IP eficiente. (s/f). Recuperado el 1 de julio de 2025, de https://efficientip.com/glossary/what-is-dnssec/

Ron Aitchison. (2011). *Pro DNS and BIND* (Apress, Ed.). www.apress.com

S. Mekala, K. D. and D. K. (2024). *"DNS DDoS Amplification Attack Detection Using Multi-Layer Perceptron Classification Algorithm,"*. IEEE. https://doi.org/10.1109/AIC61668.2024.10730978.

Southam, M. (2014). DNSSEC: What it is and why it matters. *Network Security*, *2014*(5). https://doi.org/10.1016/S1353-4858(14)70050-9

Tanenbaum, A. S. ., Wetherall, D. J. ., & Romero Elizondo, A. Vidal. (2012). *Redes de computadoras*. Pearson Educación.

Teleamazonas. (2023). *Ecuador registró más de 12 millones de ciberataques en 2023, según estudio*. https://www.teleamazonas.com/ecuador-recibio-ataques-cibeneticos-estudio/

ÍNDICE DE ANEXOS

A continuación, se listan todos los anexos incluidos en el presente trabajo de titulación:

Tabla 38

Índice de anexos del trabajo de titulación.

Anexo	Contenido
Anexo A	Topología y direccionamiento de la red
Anexo B	Configuración de los servidores DNS
Anexo C	Scripts del nodo atacante (192.168.10.66)
Anexo D	Scripts de medición
Anexo E	Cumplimiento del NIST SP 800-81r3
Anexo F	RFCs aplicados en la implementación
Anexo G	Muestras de datos de medición

Anexo A. Topología y direccionamiento de la red

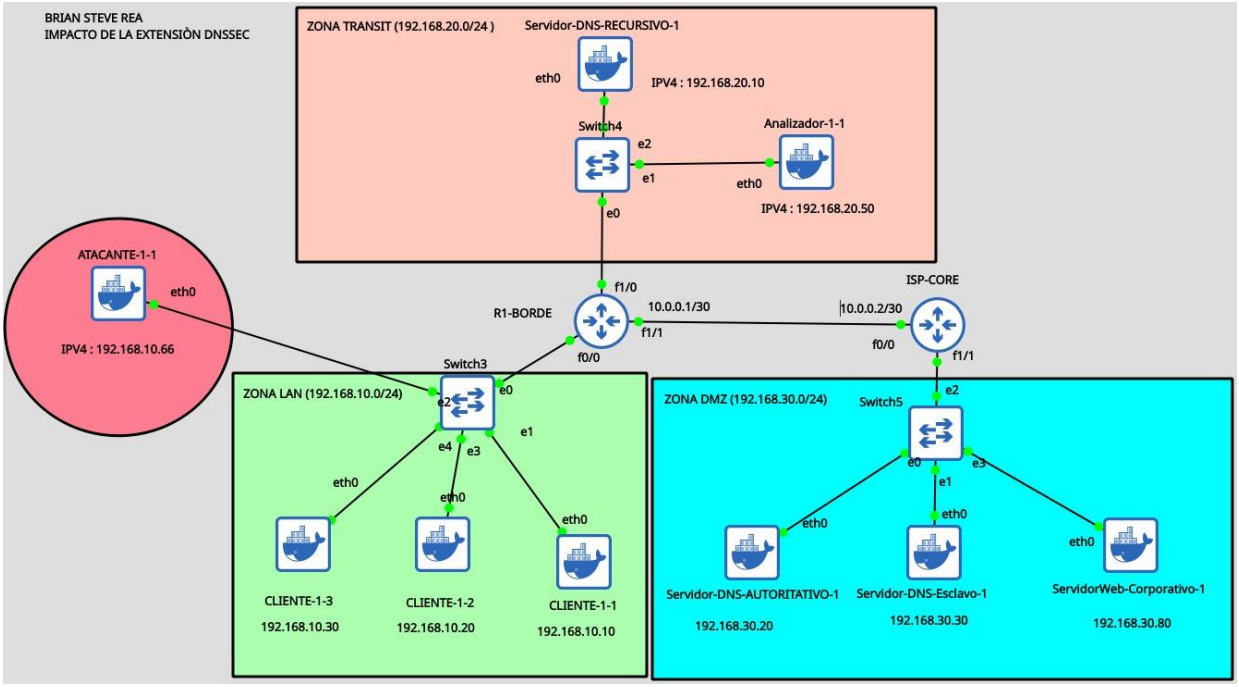
Este anexo documenta la arquitectura de red del entorno experimental, implementada sobre el emulador GNS3 con contenedores Docker y routers Cisco C7200 emulados mediante Dynamips. La topología se organiza en tres zonas de seguridad diferenciadas, interconectadas mediante enrutamiento estático, replicando la segmentación típica de una red corporativa con zona desmilitarizada (DMZ).

A.1 Diagrama de la topología

La Figura A.1 presenta la topología completa del entorno, con la distribución de los nodos en las zonas LAN, TRANSIT y DMZ, y los dos routers que interconectan los segmentos.

Figura 52

Topología de red del entorno experimental en GNS3.



Anexo B. Configuración de los servidores DNS

Este anexo recopila la configuración de los dos servidores BIND 9.20.23 desplegados en contenedores Docker dentro del entorno GNS3: el servidor autoritativo (zona empresa.local firmada con DNSSEC) y el servidor recursivo (resolución con validación DNSSEC).

B.1 Servidor DNS Autoritativo (192.168.30.20)

Servidor maestro de la zona empresa.local. Sirve la zona firmada db.empresa.local.signed; la validación DNSSEC se realiza en el recursivo, no aquí (dnssec-validation no).

Bloque de código 15

Configuración de named.conf

/etc/bind/named.conf

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
```

Bloque de código 16

Configuración de named.conf.options en el Servidor Autoritativo.

/etc/bind/named.conf.options

```
options {
    directory "/var/cache/bind";
    listen-on { any; };
    listen-on-v6 { none; };
    allow-query {
        192.168.10.0/24;
        192.168.20.0/24;
        192.168.30.0/24;
        10.10.10.0/24;
        127.0.0.1;
    };
    allow-transfer { 192.168.30.30; };
    notify yes;
    also-notify { 192.168.30.30; };
    recursion no;
    dnssec-validation no;
    minimal-responses yes;
};
logging {
    channel query_log {
        file "/var/log/named/queries.log" versions 3 size 20m;
        print-time yes; severity info;
    };
    channel default_log {
        file "/var/log/named/named.log" versions 3 size 10m;
        print-time yes; severity dynamic;
    };
    category queries  { query_log; };
    category default  { default_log; };
};
```

Bloque de código 17

Configuración del codigo named.conf.local

/etc/bind/named.conf.local

```
zone "empresa.local" {
    type master;
    file "/etc/bind/zones/db.empresa.local.signed";
```

```
    allow-update { none; };
};

zone "30.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/zones/db.192.168.30";
    allow-update { none; };
};
```

B.1.1 Archivo de zona empresa.local

Fragmento inicial del archivo de zona (227 líneas en total, 85 hosts con registros A en IPv4).

Bloque de código 18

Configuración de zonas db.empresa.local

/etc/bind/zones/db.empresa.local (primeras 50 líneas)

```
; ================================================================
;  ZONA empresa.local — Infraestructura Corporativa Completa
;  85 registros de host (solo IPv4)
; ================================================================

$TTL 300
@   IN  SOA dns-autoritativo.empresa.local. admin.empresa.local. (
            2026062301 3600 1800 604800 300 )

; === SERVIDORES DE NOMBRES ===
@               IN  NS   dns-autoritativo.empresa.local.
@               IN  NS   dns-esclavo.empresa.local.

; === INFRAESTRUCTURA DNS ===
dns-autoritativo IN  A    192.168.30.20
dns-esclavo      IN  A    192.168.30.30
dns-recursivo    IN  A    192.168.20.10

; === SERVIDORES WEB ===
web-corporativo  IN  A    192.168.30.80
www              IN  CNAME web-corporativo
webmail          IN  A    192.168.30.81
portal           IN  A    192.168.30.82
landing          IN  A    192.168.30.83
tienda           IN  A    192.168.30.84

; === CORREO Y COMUNICACIONES ===
correo           IN  A    192.168.30.85
mail             IN  A    192.168.30.85
smtp             IN  A    192.168.30.86
imap             IN  A    192.168.30.86
chat             IN  A    192.168.30.87
videoconf        IN  A    192.168.30.88
pbx              IN  A    192.168.30.89
```

B.1.2 Claves DNSSEC

Par de claves ECDSAP256SHA256 (algoritmo 13): KSK (key-signing key, ID 28475) y ZSK (zone-signing key, ID 54032).

Bloque de código 19

Claves de KSK y ZSK

```
/etc/bind/keys/*.key
```

```
        Kempresa.local.+013+28475.key    Kempresa.local.+013+28475.private    Kempresa.local.+013+54032.key
Kempresa.local.+013+54032.private
        ; This is a key-signing key, keyid 28475, for empresa.local.
        ; Created: 20260625065447 (Thu Jun 25 01:54:47 2026)
        ; Publish: 20260625065447 (Thu Jun 25 01:54:47 2026)
        ; Activate: 20260625065447 (Thu Jun 25 01:54:47 2026)
        empresa.local.  IN  DNSKEY  257  3  13  fVF1XYw/EPdURTNQFdbLtxNC/kwqrnwLJwnuJb/4nH7ReVjLkyf4WSF0
iib8gSgEabPTtuvzf19p51Yp7xVqaQ==
        ; This is a zone-signing key, keyid 54032, for empresa.local.
        ; Created: 20260625065447 (Thu Jun 25 01:54:47 2026)
        ; Publish: 20260625065447 (Thu Jun 25 01:54:47 2026)
        ; Activate: 20260625065447 (Thu Jun 25 01:54:47 2026)
        empresa.local.  IN  DNSKEY  256  3  13  2DZdc1DxVRgMX9hOKoOEZxsUUyw3BQ7dqH+5qgeilpDgSThE//e9cEsT
RdHqSJwEA7OGxILxq9ZFy8JMsIxMKg==
```

B.1.3 Estadísticas de la zona firmada

Bloque de código 20

Estadísticas de la zona firmada.

```
        RRSIG: 371

        DNSKEY: 5

        NSEC3: 187
```

B.2 Servidor DNS Recursivo (192.168.20.10)

Servidor recursivo con validación DNSSEC activada (dnssec-validation yes) y ancla de confianza estática (static-key) para la zona privada empresa.local, ya que esta no encadena con la raíz pública. Entre escenarios, lo único que cambia en este nodo es la presencia o ausencia del bloque trust-anchors y el valor de dnssec-validation.

Bloque de código 21

Configuración del archivo named.conf del Servidor Recursivo

/etc/bind/named.conf

```
include "/etc/bind/named.conf.options";

include "/etc/bind/named.conf.local";
```

Bloque de código 22

Configuración del Archivo named.conf.options del Servidor Recursivo.

/etc/bind/named.conf.options

```
options {
    directory "/var/cache/bind";
    listen-on { any; };
    listen-on-v6 { any; };
    recursion yes;
    allow-query { 192.168.10.0/24; 192.168.20.0/24; 127.0.0.1; ::1; fd10::/16; };
    allow-recursion { 192.168.10.0/24; 192.168.20.0/24; 127.0.0.1; ::1; fd10::/16; };
    dnssec-validation yes;
    forwarders { 8.8.8.8; };
    forward only;
    max-cache-size 64m;
    max-cache-ttl 300;
};
trust-anchors {
    "empresa.local."               static-key               257               3               13
"fVF1XYw/EPdURTNQFdbLtxNC/kwqrnwLJwnuJb/4nH7ReVjLkyf4WSF0iib8gSgEabPTtuvzf19p51Yp7xVqaQ==";
};
```

Bloque de código 23

Configuración del archivo named.conf.local del Servidor Recursivo.

/etc/bind/named.conf.local

```
zone "empresa.local" {
    type forward;
    forwarders { 192.168.30.20; 192.168.30.30; };
    forward only;
};
zone "30.168.192.in-addr.arpa" {
    type forward;
    forwarders { 192.168.30.20; };
    forward only;
};
```

Anexo C. Scripts del nodo Atacante (192.168.10.66)

El nodo atacante ejecuta el modelo de ataque determinista adoptado en este trabajo, compuesto por dos elementos: un servidor DNS falso que actúa como autoritativo malicioso (dns_falso.py), empleado para demostrar la aceptación o el rechazo de respuestas forjadas (componentes E3b y E4b); y un generador de carga adversaria (generar_ataque.sh), empleado para evaluar el rendimiento bajo estrés (componentes E3a y E4a). Este modelo sustituye a los enfoques de envenenamiento off-path por carrera de identificadores (Kaminsky) y de intercepción en capa 2 (ARP spoofing), descartados por su inviabilidad o por operar fuera del alcance del estudio en modo forward.

C.1 dns_falso.py - Servidor autoritativo malicioso

Servidor DNS implementado con un socket UDP en el puerto 53 que responde a las consultas de la zona empresa.local con la dirección del atacante (192.168.10.66) en lugar de la legítima. Al no incluir registros RRSIG, sus respuestas son aceptadas por el recursivo sin validación (E3b) y rechazadas con SERVFAIL cuando la validación DNSSEC está activa (E4b).

Bloque de código 24

Archivo con el script del dns_falso.py

```
#!/usr/bin/env python3
# dns_falso.py — Autoritativo FALSO sin firmar para empresa.local
# Demuestra el contraste de validacion DNSSEC (E3b acepta / E4b SERVFAIL).
# Usa un socket UDP real en el puerto 53; Scapy solo parsea/construye DNS.

import socket
from scapy.all import DNS, DNSRR, DNSQR

FORJADO = { "web-corporativo.empresa.local.": "192.168.10.66" }
IP_GENERICA = "192.168.10.66"
ZONA = "empresa.local"
PUERTO = 53
BIND_IP = "0.0.0.0"

def construir_respuesta(data):
    dns = DNS(data)                       # parsear la consulta
    if dns.qr != 0 or dns.qdcount < 1:
        return None
```

```
    qname = dns[DNSQR].qname.decode()
    qtype = dns[DNSQR].qtype
    if ZONA not in qname:
        return None
    if qtype != 1:                          # solo tipo A
        resp = DNS(id=dns.id, qr=1, aa=1, ra=0, rcode=0, qd=dns.qd)
        return bytes(resp)
    ip_resp = FORJADO.get(qname, IP_GENERICA)
    resp = DNS(id=dns.id, qr=1, aa=1, ra=0, rcode=0, qd=dns.qd,
               an=DNSRR(rrname=qname, type="A", ttl=60, rdata=ip_resp))
    print(f"[FORJADO] {qname} -> {ip_resp}")
    return bytes(resp)                      # NO incluye RRSIG

def main():
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    sock.bind((BIND_IP, PUERTO))
    print(f"[*] DNS FALSO escuchando UDP:{PUERTO} para zona {ZONA}")
    while True:
        data, addr = sock.recvfrom(4096)
        try:
            resp = construir_respuesta(data)
            if resp:
                sock.sendto(resp, addr)
        except Exception as e:
            print(f"[!] error de {addr}: {e}")

if __name__ == "__main__":
    main()
```

C.2 generar_ataque.sh - Generador de carga adversaria

Genera un archivo con 500 nombres aleatorios inexistentes de la zona empresa.local y lanza un flujo sostenido de consultas mediante dnsperf hacia el servidor recursivo. Cada nombre, al no existir, fuerza un fallo de caché que obliga al recursivo a reenviar la consulta al autoritativo, congestionando la ruta de resolución (componentes E3a y E4a).

Bloque de código 25

Archivo generar_ataque

```
#!/bin/bash
# generar_ataque.sh — Carga adversaria (E3a/E4a)
# Correr dentro del contenedor ATACANTE.


RECURSIVO_IP="192.168.20.10"
ARCHIVO="/tmp/ataque.txt"
QPS=5000            # tasa objetivo (dnsperf ajusta a la capacidad real)
CLIENTES=20
DURACION=300
```

```
# Generar 500 nombres aleatorios inexistentes
for i in $(seq 1 500); do
    echo "$(head -c8 /dev/urandom | md5sum | cut -c1-8).empresa.local A"
done > "$ARCHIVO"


# Lanzar la carga adversaria
dnsperf -s "$RECURSIVO_IP" -d "$ARCHIVO" -c "$CLIENTES" -Q "$QPS" -l "$DURACION"
```

Anexo D. Scripts de medición.

Scripts empleados desde el host para la medición estandarizada de latencia, throughput, consumo de recursos y verificación del estado de la caché.

D.1 medir.sh - Latencia, throughput y recursos

Mide la latencia en estados de caché frío (tras reiniciar named) y caliente, el throughput con dnsperf a 100, 150 y 300 QPS durante 30 segundos cada nivel, y el consumo de recursos con docker stats. Los resultados se almacenan en archivos por escenario.

Bloque de código 26

Scripts para medir latencia, throughput y recursos.

/opt/scripts/medir.sh

```
#!/bin/bash
# medir.sh — Medicion de latencia y throughput por escenario
# Uso: bash medir.sh E3
SUF="4ea40237-00ee-4702-8a98-96b3a40cd57e"
CLIENTE="GNS3.CLIENTE-1-1.$SUF"
RECURSIVO="GNS3.Servidor-DNS-RECURSIVO-1.$SUF"
ANALIZADOR="GNS3.Analizador-1-1.$SUF"
RECURSIVO_IP="192.168.20.10"
ESC="${1:-E1}"; OUT="$HOME/tesis-resultados/$ESC"; mkdir -p "$OUT"
DOMINIOS="web-corporativo webmail portal correo intranet wiki cloud gitlab \
  erp crm nomina vpn ldap grafana mysql api cdn microservicio1 load-balancer tienda"
# Latencia fria (reiniciar named)
docker exec "$RECURSIVO" sh -c 'kill $(pgrep named); named -u bind'; sleep 3
for d in $DOMINIOS; do
  docker exec "$CLIENTE" dig @$RECURSIVO_IP $d.empresa.local \
    | grep 'Query time' | awk -v x=$d '{print x,$4}'
done | tee "$OUT/latencia_fria.txt"
# Latencia caliente (repeticion)
for d in $DOMINIOS; do
  docker exec "$CLIENTE" dig @$RECURSIVO_IP $d.empresa.local \
    | grep 'Query time' | awk -v x=$d '{print x,$4}'
done | tee "$OUT/latencia_caliente.txt"
# Throughput 100/150/300 QPS, 30s
for Q in 100 150 300; do
  docker exec "$ANALIZADOR" dnsperf -s $RECURSIVO_IP -d /tmp/nombres.txt -Q $Q -l 30
```

```
        done | tee "$OUT/throughput.txt"
        # Recursos
        for i in 1 2 3 4 5; do docker stats --no-stream "$RECURSIVO" | grep RECURSIVO; sleep 2; done \
  | tee "$OUT/recursos.txt"
```

D.2 verificar-cache.sh - Monitor del estado de la caché

Monitorea la respuesta del recursivo para web-corporativo.empresa.local, indicando si la dirección devuelta es la legítima (192.168.30.80), la forjada (192.168.10.66) o si la respuesta es rechazada (SERVFAIL). Evidencia el comportamiento de E3b y E4b.

Bloque de código 27

Archivo .sh del Monitor del estado de caché.

```
        #!/bin/bash
        # verificar-cache.sh — Estado de la cache DNS del recursivo
        SUF="4ea40237-00ee-4702-8a98-96b3a40cd57e"
        CLIENTE="GNS3.CLIENTE-1-1.$SUF"; RECURSIVO_IP="192.168.20.10"
        LEGITIMA="192.168.30.80"; ATACANTE="192.168.10.66"
        while true; do
          TS=$(date +%H:%M:%S)
          RESP=$(docker exec "$CLIENTE" dig @$RECURSIVO_IP web-corporativo.empresa.local +short
2>/dev/null | head -1)
          STATUS=$(docker  exec  "$CLIENTE"  dig  @$RECURSIVO_IP  web-corporativo.empresa.local
2>/dev/null | grep -oP "status: \K[A-Z]+")
          if [ "$RESP" = "$LEGITIMA" ]; then echo "[$TS] LEGITIMA -> $RESP ($STATUS)"
          elif [ "$RESP" = "$ATACANTE" ]; then echo "[$TS] ENVENENADA -> $RESP ($STATUS)"
          elif [ "$STATUS" = "SERVFAIL" ]; then echo "[$TS] RECHAZADA -> SERVFAIL"
          else echo "[$TS] otro -> '$RESP' ($STATUS)"; fi
          sleep 2
        done
```

Anexo E. Cumplimiento del NIST SP 800-81r3

La siguiente tabla documenta la alineación de la implementación con las recomendaciones del NIST Special Publication 800-81 Revision 3 (Secure Domain Name System Deployment Guide).

Tabla 39

Tabla de cumplimiento del NIST SP 800-81r3

Ref.	Directriz	Implementación	Evidencia	Cumplimiento
§3.1	DNSSEC en autoritativos para las zonas administradas	Zona empresa.local firmada con ZSK + KSK (ECDSAP256SHA256)	371 registros RRSIG en la zona firmada	Cumple
§3.2	Algoritmos modernos; ECDSA sobre RSA	Algoritmo 13 (ECDSAP256SHA256); firmas de 64 bytes	Respuestas de 211 bytes (frente a 400+ con RSA); +107 % vs. sin DNSSEC	Cumple
§3.3	Validación DNSSEC en los recursivos	dnssec-validation yes + trust-anchors static-key	Flag AD presente en E2 y E4 (dig +dnssec)	Cumple
§3.4	NSEC3 contra enumeración de zona	Firma con -3 (NSEC3, RFC 5155); salt aleatorio	186 registros NSEC3 y 3 NSEC3PARAM en la zona	Cumple

§4.1	Separación de roles ZSK y KSK	ZSK (flag 256) firma registros; KSK (flag 257) firma el conjunto DNSKEY	Dos pares de claves verificables (28475 / 54032)	Cumple
§4.3	Rotación periódica de claves	No implementada (entorno de laboratorio)	Limitación documentada del entorno	Parcial
§6.1	Monitoreo de logs de BIND para fallos de validación	Logs de query y default con rotación	Eventos SERVFAIL registrados en E4	Cumple

Anexo F. RFC's aplicados en la implementación.

La siguiente tabla documenta los RFCs (Request for Comments) del IETF que se aplican directamente en la implementación experimental, indicando su función específica y el componente del proyecto donde se utilizan.

Tabla 40

RFC's en la implementación.

RFC	Título	Aplicación	Componente
1034	Domain Names - Concepts and Facilities	Jerarquía de la zona, SOA, NS, delegación	Autoritativo

1035	Domain Names - Implementation and Specification	Formato de mensajes DNS, tipos de registro	Autoritativo, Recursivo
4033	DNS Security Introduction and Requirements	Marco conceptual de DNSSEC	Marco teórico
4034	Resource Records for DNS Security Extensions	Registros RRSIG, DNSKEY, DS, NSEC	Autoritativo (firma)
4035	Protocol Modifications for DNSSEC	Validación de RRSIG, flag AD	Recursivo (validación)
5155	DNSSEC Hashed Authenticated Denial of Existence	NSEC3 y NSEC3PARAM contra enumeración	Autoritativo (firma)
5011	Automated Updates of DNSSEC Trust Anchors	Justificación de static-key para zona privada	Recursivo (trust anchor)
6781	DNSSEC Operational Practices, Version 2	Separación ZSK/KSK, procedimientos de firma	Implementación
6841	Framework for DNSSEC Policies	Validación de prácticas operativas	Validación

	and Practice Statements		
8080	EdDSA / curvas elípticas para DNSSEC	Referencia para algoritmos de curva elíptica	Selección de algoritmo
RFC	Título	Aplicación	Componente
1034	Domain Names - Concepts and Facilities	Jerarquía de la zona, SOA, NS, delegación	Autoritativo

La implementación aplica directamente 10 de los 12 RFCs listados. Los RFCs 5011 y BCP 38 se incluyen como referencia: el primero para justificar la decisión de usar static-key en lugar de initial-key, y el segundo como medida complementaria recomendada pero no implementada en el entorno de laboratorio.