XVII JORNADAS STIC CCN-CERT

V JORNADAS DE CIBER_DEFENSA: ESPDEF-CERT

Evolucionando la Evaluación Criptográfica – Episodio II

¿QUIÉNES SOMOS?

Sobre mí

Juan Martínez Romero

jtsec Beyond IT Security

jmartinez@jtsec.es

- Ingeniero en Tecnologías de Telecomunicación (Universidad de Granada)
- Máster en Ciberseguridad (Universidad de Granada)
- Crypto Manager & Senior Cibersecurity Consultant
- Experto en FIPS 140-2, FIPS 140-3 y PCI-PTS, entre otras metodologías.
- CriptoCert Certified Crypto Analyst

Sobre nosotros

- Servicios de evaluación y consultoría en ciberseguridad
- Laboratorio acreditado Common Criteria, LINCE y ETSI EN 303 645
- Desarrolladores de la herramienta más avanzada para Common Criteria, CCToolbox
- Implicados en actividades de estandarización (ISO, CEN/CENELEC, ISCI WGs, ENISA CSA WGs, CCUF, CMUF, ERNCIP, …)
- Miembros del SCCG (Stakeholder Cybersecurity Certification Group)
- jtsec forma parte del grupo Applus+ junto con Lightship Security. Disponemos de laboratorios en Canadá, EEUU y España

ÍNDICE

ÍNDICE

EVALUACIÓN CRIPTOGRÁFICA ACTUAL

EUROPA

- SOG-IS Crypto Evaluation Scheme Harmonised Cryptographic Evaluation Procedures v0.16 (Diciembre de 2020)
- Primera metodología de evaluación de SOG-IS
  - Implementación de mecanismos criptográficos
  - Requisitos de prevención de errores comunes de implementación

SOG-IS HEP

- SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms v1.3 (Febrero de 2023)
  - Mecanismos Criptográficos aprobados y recomendados por SOG-IS
    - Nivel aceptable de seguridad
    - Directrices de implementación

SOG-IS ACM

EVALUACIÓN CRIPTOGRÁFICA ACTUAL

ESPAÑA

- Mecanismos criptográficos autorizados por el CCN
- Incluye nuevos algoritmos autorizados por el CCN con respecto al ACM europeo
- Guía de uso transversal no limitada al ENS

Guía CCN-STIC 221

- Implementación criptográfica de referencia del CCN
- Empleada para realizar pruebas de conformidad de mecanismos criptográficos, como parte de las evaluaciones criptográficas

Biblioteca Criptográfica Botan-CCN

- Módulo de evaluación criptográfica dentro de la metodología LINCE
- Pruebas de conformidad criptográfica muy ligeras
- Enfoque de Perfiles de Protección NIAP

LINCE – MEC

jtsec Applus+

CCN centro criptológico nacional

EVALUACIÓN CRIPTOGRÁFICA ACTUAL

PROBLEMAS DETECTADOS

Fallos Típicos de Implementación

- Repetición de Vectores de Inicialización (IV) o nonces en mecanismos de cifrado simétrico
- Padding Oracle Attacks a mecanismos de cifrado simétrico basados en padding (p.ej., AES-CBC)
- Limitación de la Longitud de la Clave de Salida en esquemas de derivación de clave (KDF)
- Ataque de Bleichenbacher al esquema de criptografía asimétrica RSA-PKCSv1.5
- …

Ciclo de Vida de las Claves Criptográficas (o parámetros sensibles de seguridad)

- Generación de claves criptográficas empleando mecanismos autorizados
- Entrada y Salida segura de claves criptográficas empleando mecanismos autorizados
- Almacenamiento seguro de claves criptográficas empleando mecanismos autorizados
- Borrado seguro de claves criptográficas después del uso
- Uso correcto de las claves criptográficas (p. ej., no repetir la misma clave en distintos mecanismos criptográficos)

EVALUACIÓN CRIPTOGRÁFICA ACTUAL

PROBLEMAS DETECTADOS

Laboratorios

- La evaluación criptológica depende de la preparación y experiencia del evaluador en criptografía
- Acostumbrados a evaluar solo KATs (Known Answer Tests) de mecanismos criptográficos
- Algunos requisitos de evaluación son muy costosos (p. ej., revisión del código fuente)

Fabricantes

- Desigual conocimiento de la criptografía y de análisis de seguridad
- Delegación en librerías criptográficas sin tener en cuenta requisitos fundamentales
- No dispone de directrices claras de implementación de la criptografía

Necesitamos que los fabricantes aporten evidencias de que han evitado estos fallos típicos

EVALUACIÓN CRIPTOGRÁFICA ACTUAL

NECESIDAD DE UNA METODOLOGÍA Y HERRAMIENTA

1. Metodología de Evaluación de Mecanismos Criptográficos (MEMC)

- Sin enfocar en la arquitectura del producto (FIPS 140-3, CCN-STIC 130)
- Facilitar una lista básica de tareas de únicamente aspectos criptográficos

OBJETIVO

- Facilitar el trabajo de un evaluador con conocimientos criptográficos no profundos
- Mismos criterios de evaluación en distintas evaluaciones y por distintos laboratorios

2. Implementaciones de Referencia

- Biblioteca Criptográfica **BOTAN-CCN**
- Incluir algoritmos listados en CCN-STIC 221
- Proporcionar conjunto de vectores de tests criptográficos

Necesidad de una herramienta de testeo estándar

3. Herramienta de Pruebas de Conformidad Criptográfica

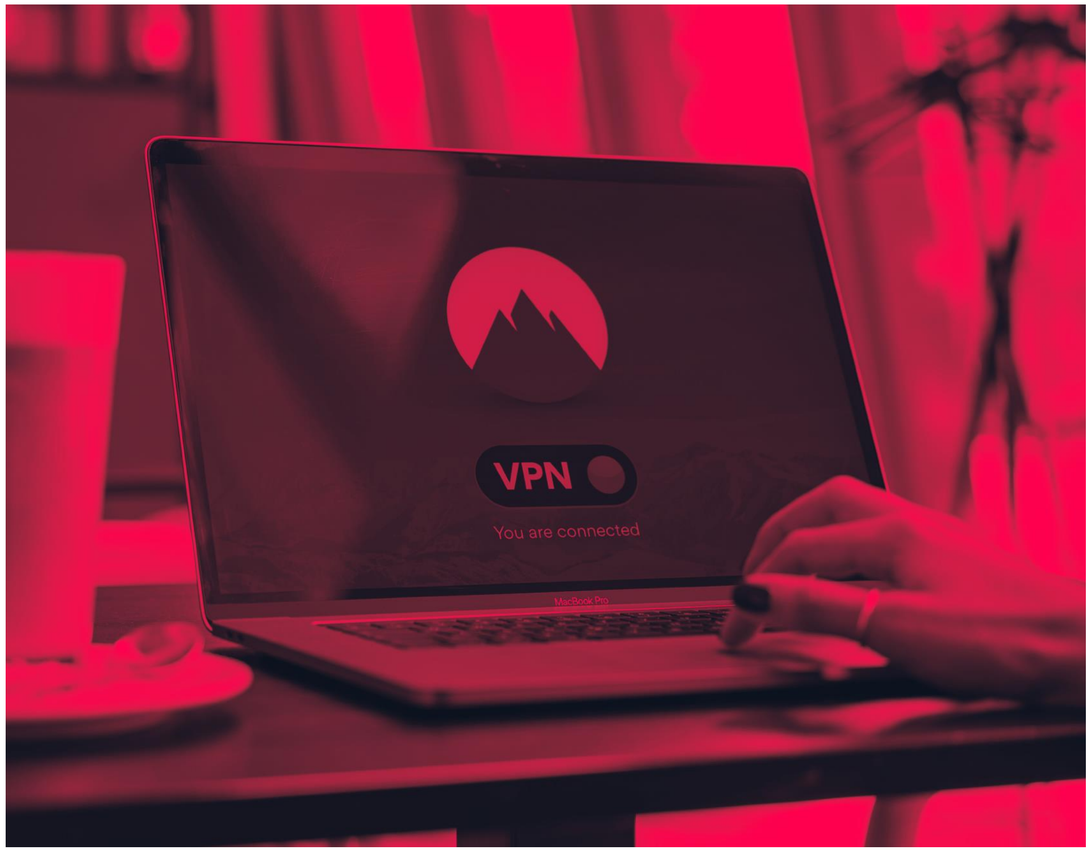
- Basada en ACVP-Parser
- Formato JSON definido en NIST ACVP → Usado en FIPS 140-3
- Implementación enlazada con BOTAN-CCN
- Posible uso por parte de la industria para acelerar el testeo de conformidad de los mecanismos criptográficos

ÍNDICE

METODOLOGÍA DE EVALUACIÓN DE MECANISMOS CRIPTOGRÁFICOS

Uso de la Metodología

- Productos que requieren de la criptografía para su funcionalidad principal (por ejemplo: VPNs, productos de cifra, comunicaciones seguras, etc.)
- De acuerdo a tres niveles de certificación ascendentes: CL1, CL2 y CL3.
- En procesos de certificación como: CC, LINCE y STIC complementarias.



METODOLOGÍA DE EVALUACIÓN DE MECANISMOS CRIPTOGRÁFICOS

Estructura de la Metodología

Índice

Estructura del Documento

- Requisitos Criptográficos
- Mecanismos Criptográficos Autorizados
- Pruebas de Conformidad
- Errores Comunes de Implementación

Proceso de Evaluación

Tareas y pruebas de evaluación

Estructura

Cada sección contiene:

- Una o varias **tareas** definidas. Estas son **obligatorias** independientemente de la implementación y se deben ejecutar para el nivel de seguridad correspondiente.

- Una o varias **pruebas** definidas correspondientes a cada tarea. Estas son clasificadas como **obligatorias** o **dependientes de la implementación**. Además, los inputs requeridos de los fabricantes son detallados en cada prueba.

TAREA DE EVALUACIÓN CRIPTOGRÁFICA	CCN-NOMBRE	NIVEL DE CERTIFICACIÓN
El **evaluador debe**... [Definición de la Tarea de Evaluación Criptográfica]		**CL1 CL2 y/o CL3**

CCN-NOMBRE/NombrePrueba	Inputs
[Definición de la Prueba de Evaluación Criptográfica]	

Proceso de Evaluación

Tareas y pruebas de evaluación

Estructura

Algunos ejemplos específicos:

TAREA DE EVALUACIÓN CRIPTOGRÁFICA	CCN-SSP	NIVEL DE CERTIFICACIÓN
El **evaluador debe** verificar que el TOE establece una gestión segura de los Parámetros Sensibles de Seguridad (SSP) durante su ciclo de vida, desde su generación hasta su destrucción, empleando mecanismos criptográficos aprobados para la generación, entrada/salida, almacenamiento y *zeroización*.		**CL2 y CL3**

Tarea de Evaluación Criptográfica	Prueba de Evaluación Criptográfica	Categoría de la Prueba
CCN-SSP Gestión de SSP	CCN-SSP/Generation	Impl-Dep
	CCN-SSP/Transport	Impl-Dep
	CCN-SSP/Storage	Obligatoria
	CCN-SSP/Zeroization.1	Obligatoria
	CCN-SSP/Zeroization.2	Obligatoria

CCN-SSP/Storage	**Inputs**
En el caso de evaluaciones **CL2**, verificar en **I1** que: - El almacenamiento de PSPs está protegido en autenticidad e integridad mediante un mecanismo criptográfico aprobado. - El almacenamiento de CSPs está protegido en autenticidad, integridad y confidencialidad mediante un mecanismo de protección de claves aprobado. Además, operar el TOE (usando **I2**) para verificar que: - Los métodos de almacenamiento de SSPs implementados por el TOE coinciden con los declarados por el fabricante en **I1**. En el caso de evaluaciones **CL3**, hacer uso de **I1**, operar el TOE (usando **I2**) y realizar revisión del código fuente (usando **I6**) para verificar lo anteriormente descrito.	**I1** **I2** **I6**

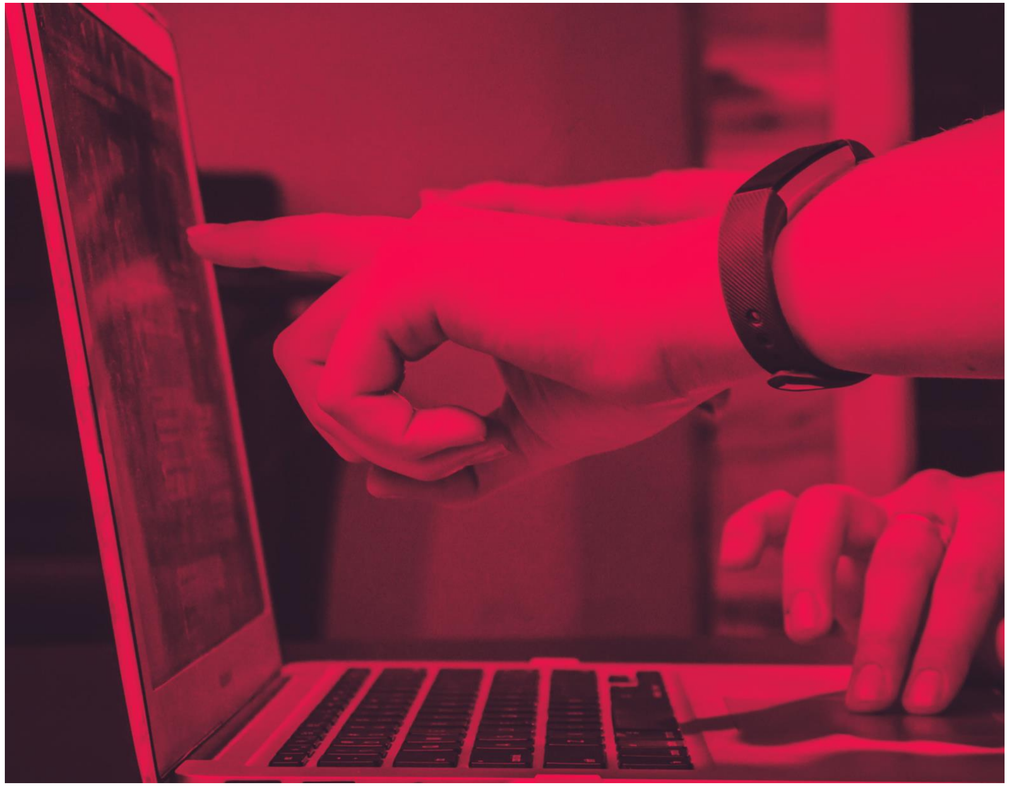
Metodología de Evaluación

1. Requisitos Criptográficos

Objetivo: Especificar los requisitos extraídos por el CCN de la guía CCN-STIC 130 que aplican a la seguridad de los productos criptográficos asociados a los mecanismos criptográficos implementados, en relación con:

- Self-tests (no exigidos por el SOG-IS ni por la CCN-STIC 221)
- Gestión de Parámetros Críticos de Seguridad (CSP) (con requisitos adicionales a los exigidos por el SOG-IS)
- Mitigación de Otros Ataques (no exigido por el SOG-IS ni por la CCN-STIC 221)

Evaluación: El evaluador deberá verificar que el TOE cumple con los requisitos criptográficos expuestos en esta sección.



Metodología de Evaluación

1. Requisitos Criptográficos

Gestión de los Parámetros Críticos de Seguridad (CSP)

La metodología no sólo evalúa los requisitos de Gestión de Claves definidos por el SOG-IS, sino que también todo el ciclo de vida de cada SSP gestionado por el TOE.

Este enfoque integral garantiza una evaluación exhaustiva desde la postura de seguridad del TOE más allá de la mera gestión de claves.

Ejemplo: Gestión del Ciclo de Vida del SSP para una clave AES

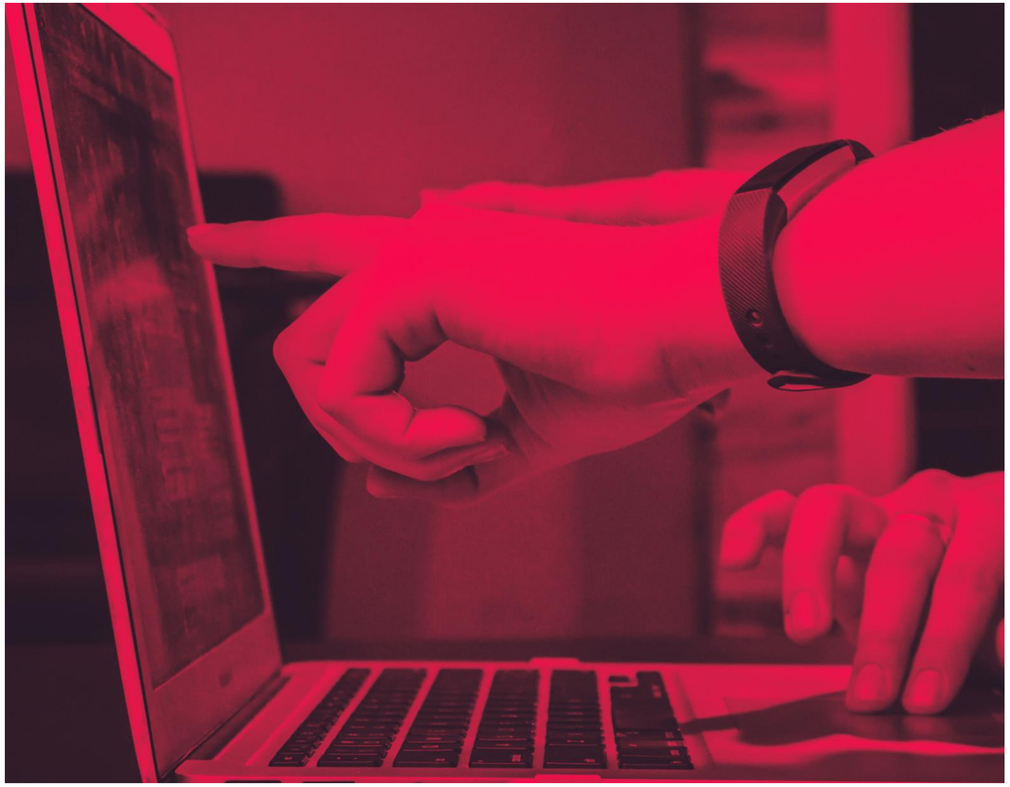
SSP	Fortaleza (en bits)	Generación	Entrada/Salida	Almacenamiento	Aplicación u Operación Criptográfica	*Zeroización*
AES_EDK	128 256	**Método** DRBG (Hash_DRBG)	**Método** N/A	**Método** AES-256 KeyWrap	**Aplicación u Operación Criptográfica** Cifrado/Descifrado con: AES-CBC AES-CTR	**Método** Sobreescritura usando ceros
		Evidencia para CL3 Incluya un fragmento de código fuente y una justificación	**Evidencia para CL3** N/A	**Evidencia para CL3** Incluya un fragmento de código fuente y una justificación	**Evidencia para CL3** Incluya un fragmento de código fuente y una justificación	**Evidencia para CL2 y CL3** Incluya un fragmento de código fuente y una justificación

Metodología de Evaluación

2. Mecanismos Criptográficos Autorizados

Objetivo: Especificar los mecanismos criptográficos reconocidos y autorizados por el CCN.

Evaluación: El evaluador deberá verificar que los mecanismos criptográficos implementados por el TOE cumplen con las directrices presentadas por el CCN en la guía CCN-STIC 221, incluyendo la correcta parametrización.



Metodología de Evaluación

2. Mecanismos Criptográficos Autorizados

Requisitos para Generadores de Números Aleatorios

Generadores de Números Realmente Aleatorios

- **Generadores Físicos (PTRNG)** → PTG.2 o PTG.3
  - Hardware dedicado que genera ruido
  - Evaluación basada en Modelo Estocástico
- **Generadores No Físicos (NPTRNG)** → NTG.1
  - Recursos del sistema para generar ruido
  - Linux */dev/random* bajo ciertas condiciones

AIS-31 (BSI)

Generadores de Números Aleatorios Deterministas

- **Generadores DRNG** → DRG.3 o DRG.4
  - Basados en Hash_DRBG, HMAC_DRBG o CTR_DRBG
  - Uso de TRNG para semillado y resemillado

AIS-20 (BSI)

Metodología de Evaluación

3. Pruebas de Conformidad

Objetivo: Especificar los requisitos necesarios para realizar las pruebas de conformidad de los mecanismos criptográficos implementados por el TOE.

Estas pruebas determinarán si las primitivas y construcciones criptográficas utilizadas por el TOE están correctamente implementadas. Esto es similar a lo que hace el NIST, pero verificando también las parametrizaciones y los valores límite que suelen dar lugar a errores.

Pruebas de Conformidad:

- Tests de Montecarlo (MCT)
- Known Answer Tests (KAT)
- Tests de Validación (VAL)

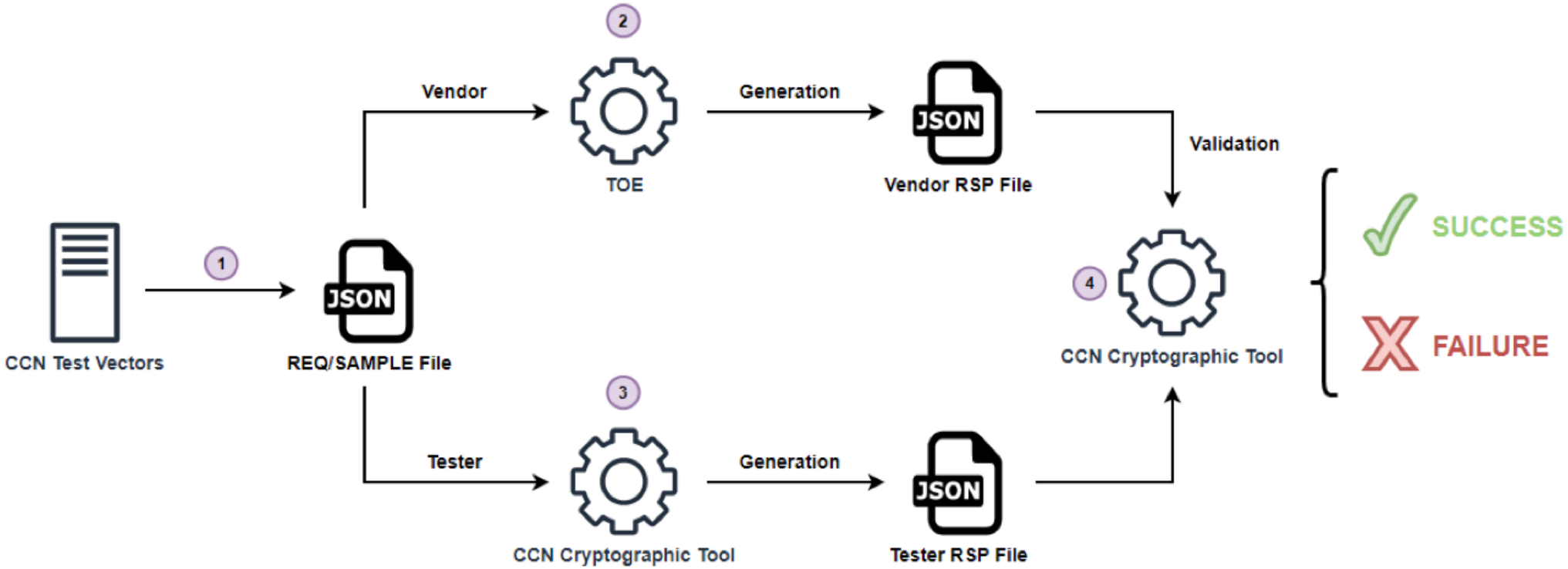
Evaluación: El proceso de evaluación se divide en 4 etapas:

1. Generación de los Vectores de Test: ficheros Request y Sample.
2. Generación de Resultados por el Fabricante: fichero Response
3. Generación de Resultados por el Evaluador: fichero Response
4. Validación de Resultados por el Evaluador

Metodología de Evaluación

3. Pruebas de Conformidad

Diagrama del Proceso de Evaluación de las Pruebas de Conformidad



Metodología de Evaluación

3. Pruebas de Conformidad

Generación de Vectores de Prueba

- El evaluador deberá generar un fichero 'REQUEST' (en formato JSON) para cada mecanismo criptográfico implementado por el TOE, conteniendo los vectores de prueba asociados a la parametrización soportada.

- Además, el evaluador deberá generar el fichero "SAMPLE" (en formato JSON) para cada mecanismo criptográfico implementado por el TOE, que contenga una solución de ejemplo para indicar el formato del resultado esperado.

- El evaluador deberá enviar al fabricante un paquete de ficheros que contenga los ficheros 'REQUEST' y 'SAMPLE' asociados a todos los mecanismos criptográficos implementados por el TOE.

Metodología de Evaluación

3. Pruebas de Conformidad

Generación de Resultados por el Fabricante

- El fabricante deberá generar un fichero 'RESPONSE' asociado a cada mecanismo criptográfico implementado, que contendrá la salida proporcionada por el TOE para cada uno de los vectores de prueba, proporcionados en el fichero 'REQUEST'.

- El fabricante deberá conservar el formato JSON presentado en los ficheros 'REQUEST' y 'SAMPLE' para la generación del fichero 'RESPONSE'.

- El fabricante deberá enviar al evaluador un paquete de ficheros que contenga los ficheros 'RESPONSE' asociados a todos los mecanismos criptográficos implementados por el TOE.

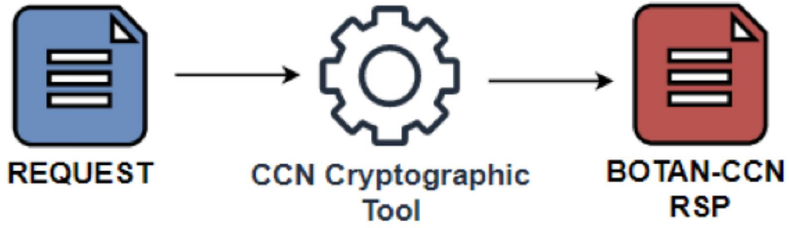
Metodología de Evaluación

3. Pruebas de Conformidad

Generación de Resultados por el Evaluador

- El evaluador deberá generar el fichero 'RESPONSE' asociado a cada mecanismo criptográfico implementado por el TOE, utilizando la biblioteca Botan-CCN como implementación criptográfica de referencia.

- El evaluador deberá conservar el formato JSON presentado en los ficheros 'REQUEST' y 'SAMPLE' para la generación del fichero 'RESPONSE'.

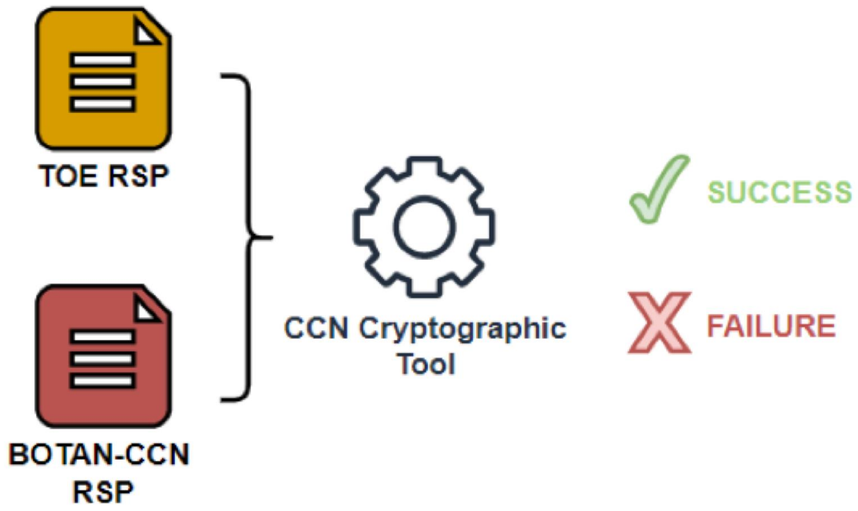


METODOLOGÍA DE EVALUACIÓN DE MECANISMOS CRIPTOGRÁFICOS

Metodología de Evaluación

3. Pruebas de Conformidad

Validación de Resultados por el Evaluador

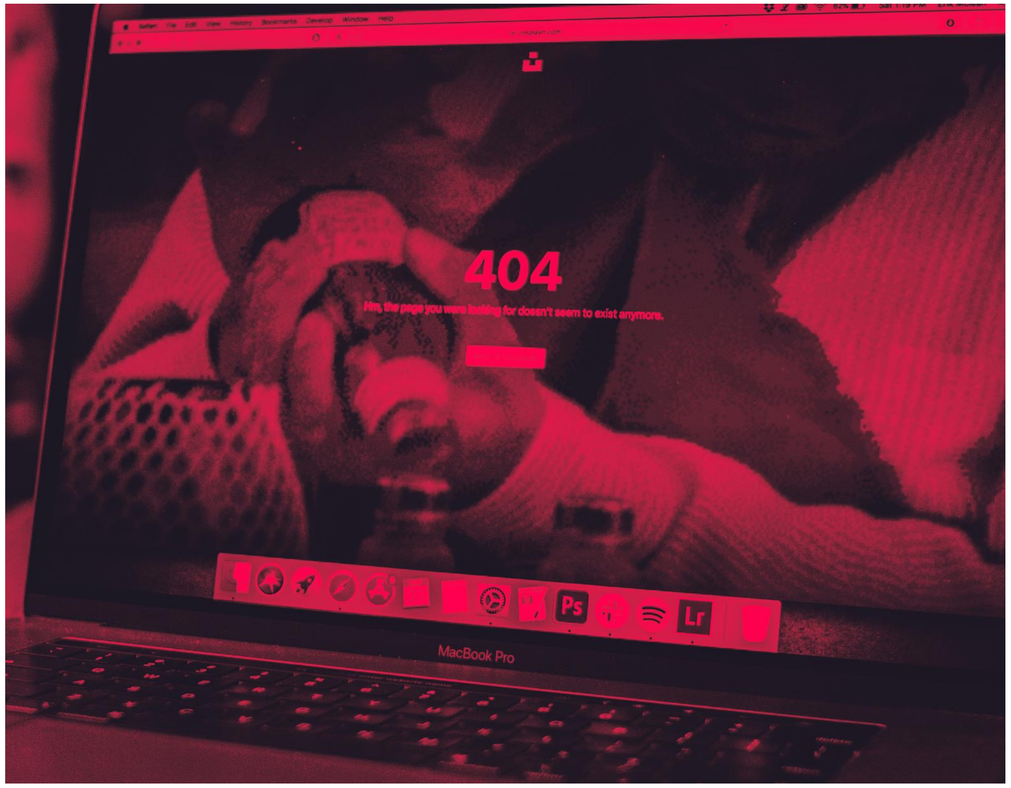


- El evaluador deberá validar los ficheros 'RESPONSE' proporcionados por el fabricante para cada mecanismo criptográfico implementado por el TOE, comparando los resultados proporcionados con los obtenidos en el paso anterior utilizando la biblioteca criptográfica Botan-CCN.

- El evaluador deberá determinar si el TOE implementa correctamente las primitivas y construcciones criptográficas utilizadas y declaradas.

Metodología de Evaluación

4. Errores Comunes de Implementación



Objetivo: Especificar los requisitos necesarios para evitar errores de implementación en los mecanismos criptográficos implementados por el TOE.

Evaluación: El evaluador deberá verificar que los mecanismos criptográficos implementados por el TOE cumplen con las directrices para evitar errores de implementación presentadas por SOG-IS en la guía SOG-IS Harmonized Cryptographic Evaluation Procedures (HEP).

Metodología de Evaluación

4. Errores Comunes de Implementación

Ejemplo: Pitfall en la Implementación de Derivación de Claves

CCN-PITFALL/KeyDerivation	Inputs
En el caso de evaluaciones **CL2**, verificar mediante **I5** que no es posible realizar peticiones invalidas para la generación de claves. **Análisis:** La computación de la clave derivada comienza con controles de tamaño que no deben ignorarse. En particular, el evaluador debe verificar que ninguna clave derivada tiene mayor tamaño que: - $255 \times h$ para las construcciones HKDF - $(2^{32} - 1) \times h$ para el resto de funciones de derivación de claves donde h es la longitud (en bits) del bloque de salida de la función hash subyacente o la función pseudoaleatoria. En el caso de evaluaciones **CL3**, utilizar **I5** y revisar el código fuente (mediante **I6**) para verificar lo anterior.	**I5** **I6**

Metodología de Evaluación

Anexo A. Trazabilidad de las Tareas de Evaluación

Objetivo: Resumir en un conjunto de tablas las tareas de evaluación requeridas para cada mecanismo criptográfico, asociadas a cada sección de la CCN MEMC.

Evaluación: El evaluador deberá realizar todas las tareas de evaluación recogidas en la tabla asociada al mecanismo criptográfico, para determinar si está correctamente implementado por el TOE.

Summary of Evaluation Tasks for GCM AE Constructions	
Evaluation Task	**Evaluation Test Code**
CCN-MECHANISM	**Construction** CCN-MECHANISM/AuthEncryption
	Underlying Primitive CCN-MECHANISM/BlockCipher
CCN-AGREED	**Construction** CCN-AGREED/AuthEncryption **Notes** Note 12 [GMAC-GCM Nonce] Note 13 [GMAC-GCM Options] Note 14 [GMAC-GCM Bounds] Note 18 [Authenticated Encryption Schemes] Note 19 [Decryption Order] Note 20 [GCM Plaintext Length]
	Underlying Primitive CCN-AGREED/BlockCipher
CCN-CONFORMITY	**Construction and Underlying Primitive** CCN-CONFORMITY/Mechanisms (Table 59)
CCN-PITFALL	**Construction** CCN-PITFALL/AuthEncryption CCN-PITFALL/GCM.1 CCN-PITFALL/GCM.2
	Underlying Primitive N/A because there are no implementation pitfalls for the AES cryptographic primitive

METODOLOGÍA DE EVALUACIÓN DE MECANISMOS CRIPTOGRÁFICOS

Ventajas de la Metodología de Evaluación de Mecanismos Criptográficos sobre la del SOG-IS

CCN MEMC

- **Metodología de Evaluación Completa.** Establece tareas concretas de evaluación en función del nivel de certificación (CL1, CL2 o CL3):
  - Requisitos de gestión de claves criptográficas
  - Mitigación de otros ataques, como ataques de canal lateral
  - Utilización de mecanismos aprobados, incluidos algoritmos postcuánticos y requisitos específicos de entropía
  - Pruebas de Conformidad
  - Evitar errores comunes de implementación (se han añadido nuevos con respecto a SOG-IS)

- **Self-tests.** Establece tareas de evaluación para verificar la implementación y correcto funcionamiento de self-tests de mecanismos criptográficos.

SOG-IS ACM & HEP

- **Mecanismos Aprobados y Requisitos.** Proporcionan los mecanismos aprobados y sus requisitos asociados, así como las tareas de evaluación para:
  - Verificar su correcta implementación según su norma asociada.
  - Realizar las pruebas de conformidad.
  - Evitar errores comunes de implementación.
  - Verificar la gestion de claves (con menos requisitos que la CCN MEMC)

No se especifican requisitos asociados a la implementación de self-tests ni a la mitigación de otros ataques

SOGIS
Senior Officials Group
Information Systems Security

jtsec Applus+

CCN centro criptológico nacional

METODOLOGÍA DE EVALUACIÓN DE MECANISMOS CRIPTOGRÁFICOS

Ventajas de la Metodología de Evaluación de Mecanismos Criptográficos sobre la del SOG-IS

CCN MEMC

Nuevos Algoritmos

Incluye los nuevos algoritmos clásicos y postcuánticos recomendados por el CCN en la nueva guía STIC 221

Nuevos algoritmos clásicos recomendados		Algoritmos Postcuánticos	
SCRYPT y ChaCha20-Poly1305	EdDSA y X25519	NIST PQC CRYSTALS-Kyber, CRYSTALS-Dilithium, Falcon y SPHINCS+	CCN PQC FrodoKEM también se recomienda por CCN

SOG-IS ACM & HEP

- Lista de algoritmos criptográficos clásicos sin incluir referencias a algoritmos postcuánticos

SOGIS
Senior Officials Group
Information Systems Security

jtsec Applus

CCN centro criptológico nacional

METODOLOGÍA DE EVALUACIÓN DE MECANISMOS CRIPTOGRÁFICOS

Ventajas de la Metodología de Evaluación de Mecanismos Criptográficos sobre la del SOG-IS

CCN MEMC

- **Gestión del Ciclo de Vida para cada SSP gestionado por el TOE.**
  - Fortaleza y Uso
  - Método de Generación
  - Método de Entrada/Salida
  - Método de Almacenamiento
  - Método de *Zeroización*

- **Lista completa de vectores de prueba de conformidad para todos los mecanismos criptográficos aprobados.**
  - Ejemplo: AES Key Wrapping.

SOG-IS ACM & HEP

- Establece los requisitos generales de Gestión de Claves, especificando solo el mecanismo recomendado para cada etapa.
- Los vectores de prueba de conformidad de varios mecanismos criptográficos no están definidos o no están completos.

SOGIS
Senior Officials Group
Information Systems Security

jtsec Applus+

CCN centro criptológico nacional

METODOLOGÍA DE EVALUACIÓN DE MECANISMOS CRIPTOGRÁFICOS

Enlace con Evaluaciones Common Criteria

Metodología de Evaluación de Mecanismos Criptográficos

- La metodología se utilizará en las evaluaciones de Common Criteria a nivel nacional si la criptografía es un componente principal del producto (p.ej., VPN, módulos de cifra, ...)

- La metodología podría considerarse un documento de apoyo para dirigir la forma de evaluar los mecanismos criptográficos.

ÍNDICE

HERRAMIENTA DE EVALUACIÓN CRIPTOGRÁFICA

Definición de la Herramienta

Estructura

- **Ficheros de Prueba JSON**

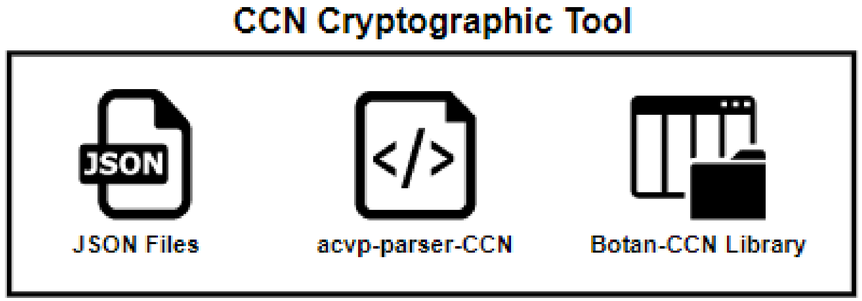
Vectores de prueba en formato hexadecimal según la metodología del SOG-IS para todos los mecanismos criptográficos. Contiene los tests para las pruebas de conformidad de tipo MCT, KAT y VAL.

- **Herramienta acvp-parser-CCN**

Herramienta encargada del procesamiento de ficheros JSON y extracción de los parámetros necesarios para invocar la implementación de referencia criptográfica, en este caso Botan-CCN.

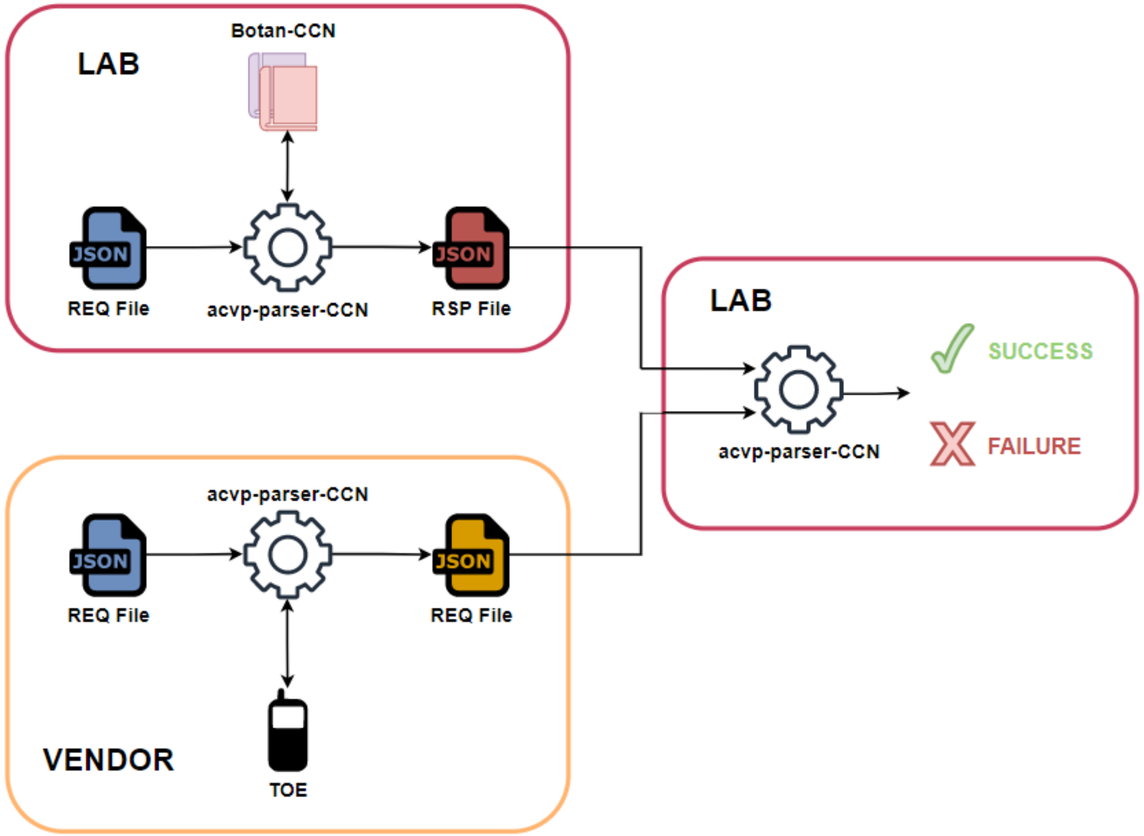
- **Biblioteca Criptográfica Botan-CCN**

Implementación de referencia criptográfica utilizada para generar resultados de vectores de prueba y poder validar la correcta implementación criptográfica del TOE.



Definición de la Herramienta

Diagrama



LABORATORIO

- Procesamiento de los vectores de prueba para extraer los parámetros mediante la herramienta acvp-parser-CCN.
- Invocación de la biblioteca criptográfica Botan-CCN para realizar la generación de los resultados del vector de prueba utilizando el fichero ' REQUEST ' asociado.
- Generación del fichero 'RESPONSE' asociado a un mecanismo criptográfico utilizando el fichero 'REQUEST' asociado y los resultados obtenidos utilizando la librería criptográfica Botan-CCN.

FABRICANTE

- Permitirá realizar el mismo proceso que en el laboratorio, pero invocando a la librería criptográfica del TOE.

Ejemplo de Uso

AES-256 CTR

AES256-CTR.req.json | AES256-CTR.rsp.json

home > kali > tests > {} AES256-CTR.req.json > ...

```
[
  {
    "Version": "1.0"
  },
  {
    "vsId": 0,
    "algorithm": "AES-CTR",
    "state": "AES Encryption and Decryption",
    "paddingScheme": "No-Padding",
    "revision": "1.0",
    "testGroups": [
      {
        "tgId": 0,
        "testType": "KAT",
        "direction": "encrypt",
        "keyLen": 256,
        "tests": [
          {
            "count": 0,
            "key": "8a205062866ab3535c4814fb6a26dd049447c9de06472f96c2c99e6aadba5c7b",
            "iv": "c1120a0113c33143538e6ea931b0d1d7",
            "plaintext": "dcabd1fae5631fe426d35113e6fb40729a",
            "ciphertext": ""
          },
```

Fichero ‘REQUEST’

AES256-CTR.req.json | AES256-CTR.rsp.json

home > kali > tests > {} AES256-CTR.rsp.json > {} 1 > [ ] testGroups > {} 0

```
[
  {
    "Version": "1.0"
  },
  {
    "vsId": 0,
    "algorithm": "AES-CTR",
    "state": "AES Encryption and Decryption",
    "paddingScheme": "No-Padding",
    "revision": "1.0",
    "testGroups": [
      {
        "tgId": 0,
        "testType": "KAT",
        "direction": "encrypt",
        "keyLen": 256,
        "tests": [
          {
            "count": 0,
            "key": "8a205062866ab3535c4814fb6a26dd049447c9de06472f96c2c99e6aadba5c7b",
            "iv": "c1120a0113c33143538e6ea931b0d1d7",
            "plaintext": "dcabd1fae5631fe426d35113e6fb40729a",
            "ciphertext": "2997859e34d17e6bc3098b28e66b853acf"
          },
```

Fichero ‘RESPONSE’ generado por la Herramienta

HERRAMIENTA DE EVALUACIÓN CRIPTOGRÁFICA

Ejemplo de Uso

AES-256 CTR

AES256-CTR.req.json | AES256-CTR.rsp.json

home > kali > tests > {} AES256-CTR.rsp.json > {} 1 > [ ] testGroups > {} 0

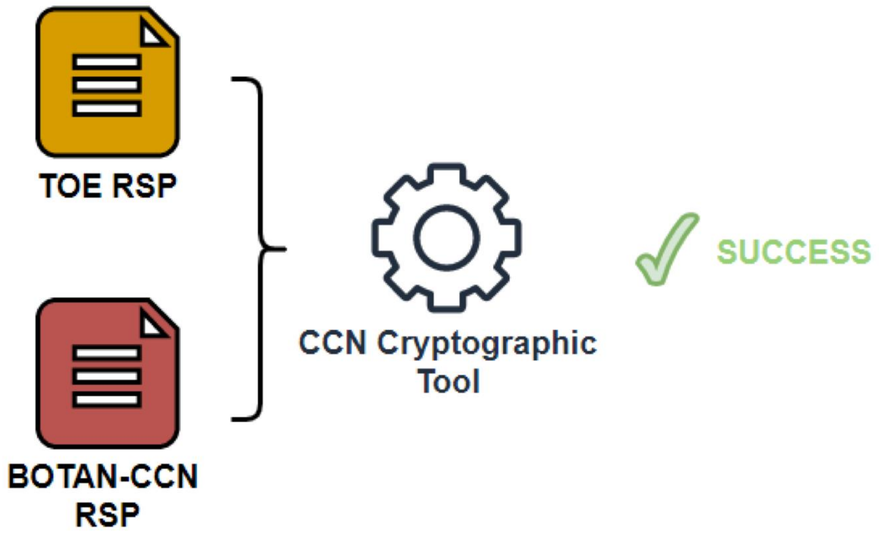
```
[
  {
    "Version": "1.0"
  },
  {
    "vsId": 0,
    "algorithm": "AES-CTR",
    "state": "AES Encryption and Decryption",
    "paddingScheme": "No-Padding",
    "revision": "1.0",
    "testGroups": [
      {
        "tgId": 0,
        "testType": "KAT",
        "direction": "encrypt",
        "keyLen": 256,
        "tests": [
          {
            "count": 0,
            "key": "8a205062866ab3535c4814fb6a26dd049447c9de06472f96c2c99e6aadba5c7b",
            "iv": "c1120a0113c33143538e6ea931b0d1d7",
            "plaintext": "dcabd1fae5631fe426d35113e6fb40729a",
            "ciphertext": "2997859e34d17e6bc3098b28e66b853acf"
          },
```

PASS

Fichero ‘RESPONSE’ generado por el TOE

```
$ ./acvp-parser -e AES256-CTR.rsp.json KNOWN-AES256-CTR.rsp.json -v
[PASSED] compare AES256-CTR.rsp.json with KNOWN-AES256-CTR.rsp.json
$
```

Validación de Resultados usando la Herramienta



HERRAMIENTA DE EVALUACIÓN CRIPTOGRÁFICA

Ejemplo de Uso

AES-256 CTR

AES256-CTR.rsp.json

home > kali > tests > {} AES256-CTR.rsp.json > {} 1 > [ ] testGroups > {} 0 > [ ] tests > {} 1


```
[
  {
    "Version": "1.0"
  },
  {
    "vsId": 0,
    "algorithm": "AES-CTR",
    "state": "AES Encryption and Decryption",
    "paddingScheme": "No-Padding",
    "revision": "1.0",
    "testGroups": [
      {
        "tgId": 0,
        "testType": "KAT",
        "direction": "encrypt",
        "keyLen": 256,
        "tests": [
          {
            "count": 0,
            "key": "8a205062866ab3535c4814fb6a26dd049447c9de06472f96c2c99e6aadba5c7b",
            "iv": "c1120a0113c33143538e6ea931b0d1d7",
            "plaintext": "dcabd1fae5631fe426d35113e6fb40729a",
            "ciphertext": "ec977fc1a287cdaaf86726819781470d4c46"
          },
```

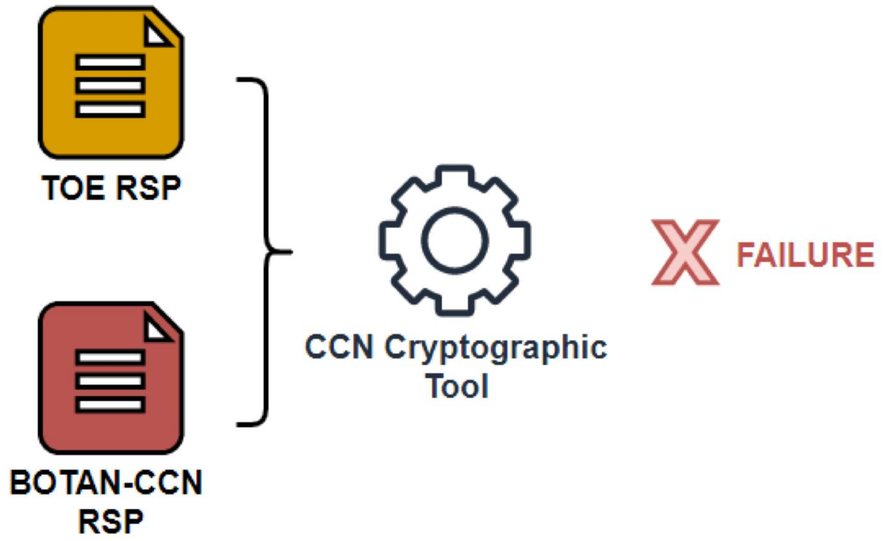

FAIL

Fichero ‘RESPONSE’ generado por el TOE

ERROR

```
$ ./acvp-parser -e AES256-CTR.rsp.json KNOWN-AES256-CTR.rsp.json -v
[FAILED] compare AES256-CTR.rsp.json with KNOWN-AES256-CTR.rsp.json
$
```

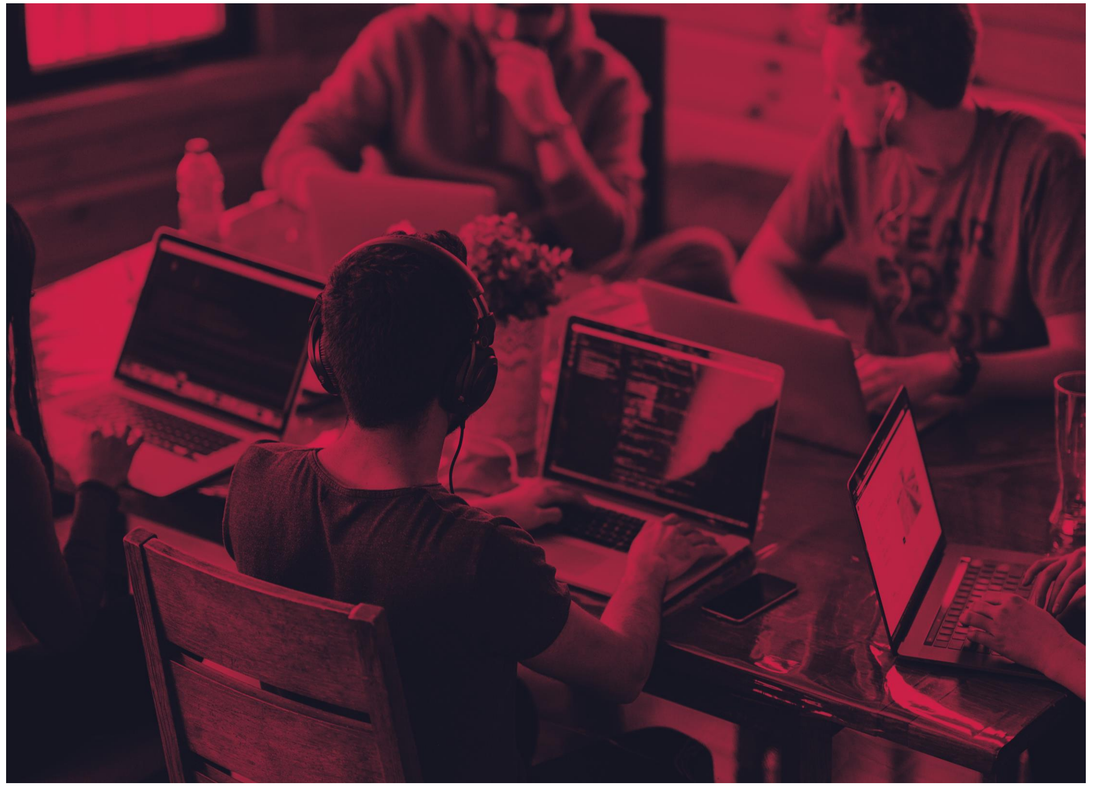
Validación de Resultados usando la Herramienta



ÍNDICE

CONCLUSIONES

- España es pionera en la creación de una Metodología de Evaluación Criptográfica **para mecanismos**.
- El **uso en las evaluaciones Common Criteria es sencillo**.
- La metodología está en **fase de prueba** y será publicada pronto.
- Todo este trabajo es una contribución para complementar los **esfuerzos europeos.**
- Este esfuerzo es necesario para **unificar criterios** en el sector con el fin de facilitar la vida a laboratorios y fabricantes.



MUCHAS GRACIAS

COMPARTIR PARA GANAR

GOBIERNO DE ESPAÑA
MINISTERIO DE DEFENSA

U 23
PRESIDENCIA ESPAÑOLA CONSEJO DE LA UNIÓN EUROPEA

CCN centro criptológico nacional

CCN-CERT centro criptológico nacional

espdef-cert
Mando Conjunto del Ciberespacio