REPORTE EJECUTIVO DE HALLAZGOS DE SEGURIDAD EN APLICACIÓN WEB

Enero 30, 2024

Versión 1.0

Resumen

Se ha llevado a cabo a cabo una evaluación técnica de seguridad sobre una aplicación web desarrollada para que intencionalmente fuera vulnerable desde diferentes vectores de ataque para poder lograr su explotación y así poder identificar los patrones de ataque, técnicamente haciendo uso de herramientas que visualicen esta información. El test se ejecutó de manera local y cada vulnerabilidad identificada ha sido documentada tanto manualmente como en una API de consulta a demanda desarrollada para este propósito, en donde puede obtenerse la descripción de cada una, remediaciones, detalles de explotación y potencial escalamiento de la misma.

Alcance

El alcance de esta prueba fue tan solo una dirección IP en una red privada interna

Dirección IP	Descripción
192.168.20.5:8000	Aplicación web local

Análisis general

La aplicación desarrollada consiste en un portal básico de noticias, el cual permite visualizar las noticias más actuales de 3 portales periodísticos muy reconocidos a nivel mundial y para el cual se requiere primero autenticarse con unas credenciales que le permita al usuario acceder al portal y poder ver el contenido, además de poder buscar por medio de una barra de búsqueda, el contenido que desee.

De acuerdo con su desarrollo y configuración, se pudieron identificar 6 vulnerabilidades susceptibles a diversos vectores de ataques web, de las cuales 2 son de severidad alta, 3 de severidad media y una de severidad baja.

Dos de los hallazgos, están relacionados con el sistema de autenticación el cual no hace una correcta validación de caracteres y permite inyectar consultas SQL en los campos que luego serán ejecutadas en la base de datos; además carece de un mecanismo de bloqueo de usuarios sin importar los intentos de autenticación que se hagan allí por lo que el sistema es propenso a sufrir ataques por medio de vectores de fuerza bruta. Otro está asociado con la falta de control de acceso de la aplicación, la cual un usuario sabiendo las rutas del sitio, puede simplemente acceder a ellas sin necesidad de autenticarse primero. Otro más está asociado con lo permisivo del sitio para crear frames y así poder crear elementos invisibles en la app que se superpongan en elementos visibles y poder engañar a quien esté usándolo. Por último, el sitio es vulnerable en su sistema de barra de búsqueda debido a que el sitio utiliza funciones inseguras que no sanitizan bien las entradas de los usuarios y permite ejecutar código.

Resumen de los hallazgos

Número de hallazgo	Severidad	Nombre	Matriz MITRE ATT&CK
1	Alta	Falta de validación de la entrada de los datos (SQLi)	T1505.001
2	Alta	Control de acceso débil	T1556
3	Media	Falta de sanitización de caracteres XSS	T1189
4	Media	Uso de funciones de template inseguras en código SSTI	T1189
5	Media	Falta de mecanismo de bloqueo de usuario	T1110.001
6	Baja	Es posible crear frames invisibles del sitio	T1189

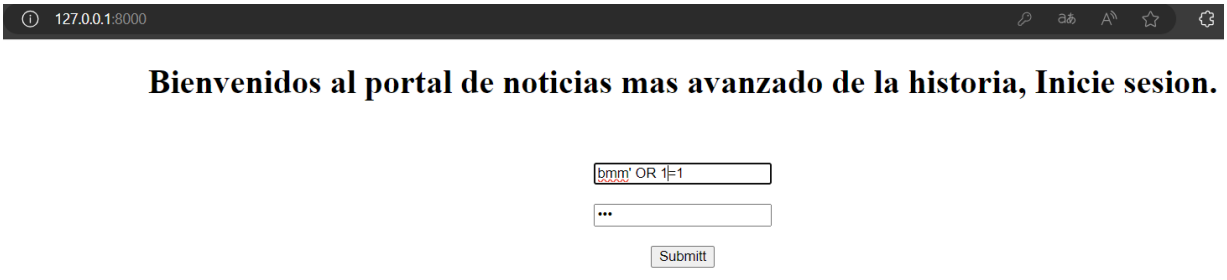
Vectores de Ataque

Los vectores de ataque son todas esas formas o metodologías por medio de las cuales se puede llevar acabo un ataque específico. Para este caso se pudieron identificar varios en el sitio web a medida que los hallazgos de vulnerabilidades iban surgiendo.

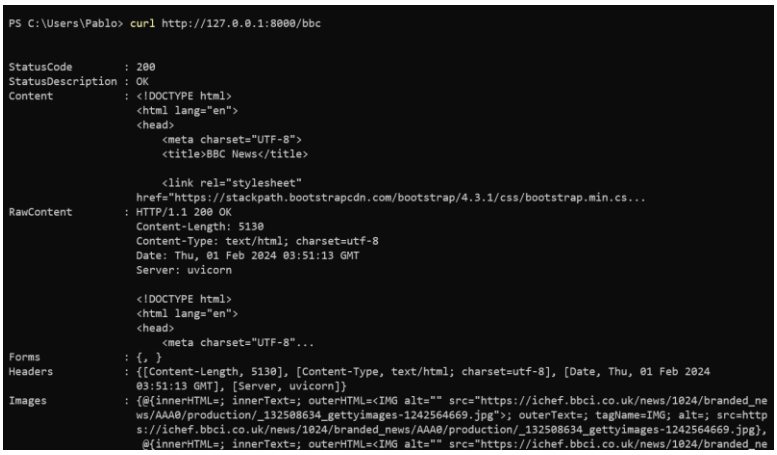
- el primer hallazgo se logra al poderse saltar el sistema de autenticación que no hace una correcta validación de entrada de los datos del usuario lo que permite hacer inyecciones de código SQL y ser ejecutadas directamente en la base de datos a conveniencia.

  Vectores:

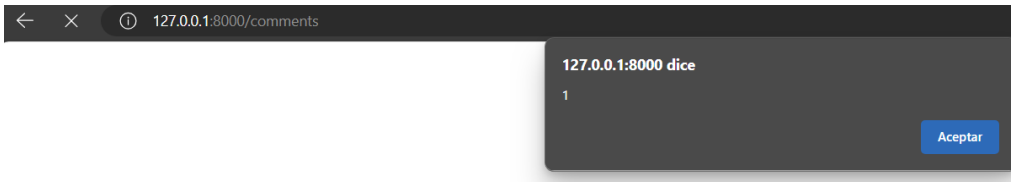
  1. Ataque de inyección SQL
  2. Ataque de denegación de servicio
  3. Ataque de fuerza bruta



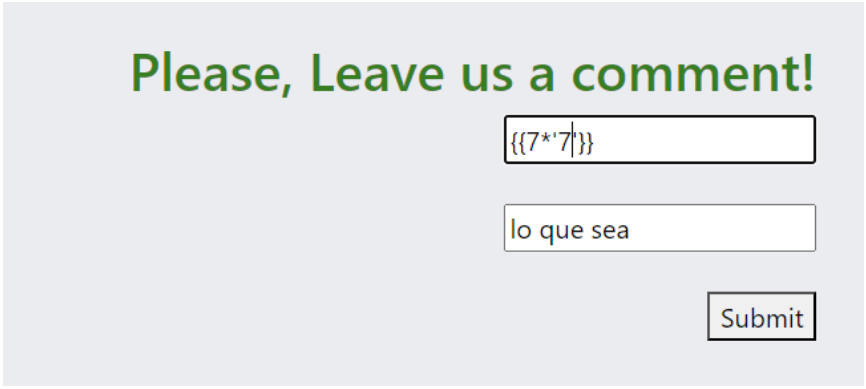
- Una vez se logra autenticarse en el sitio y navegar sobre este, es posible identificar los diferentes paths asociados a la URL. /bbc, /cnn, /abc-news. Debido a la falta de control de acceso del sitio, aun sin tener que autenticarse, es posible acceder al contenido simplemente accediendo directamente a los paths de la url.
  Vectores:
  1. Ataque de recorrido de directorios
  2. Ataque de fuerza bruta

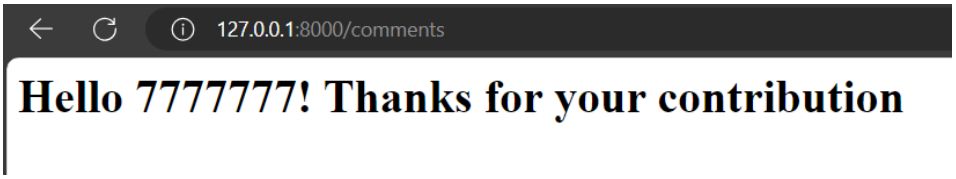


- Fue posible hacer que los datos de entrada en los campos de dejar comentario y nombre pudieran ejecutarse como código javascript en el navegador lo que lo hace vulnerable a muchos tipos de ataque de Cross Site Scripting XSS
  Vectores:
  1. Ataques de inyección XSS
  2. Ejecución remota de código
  3. DoS

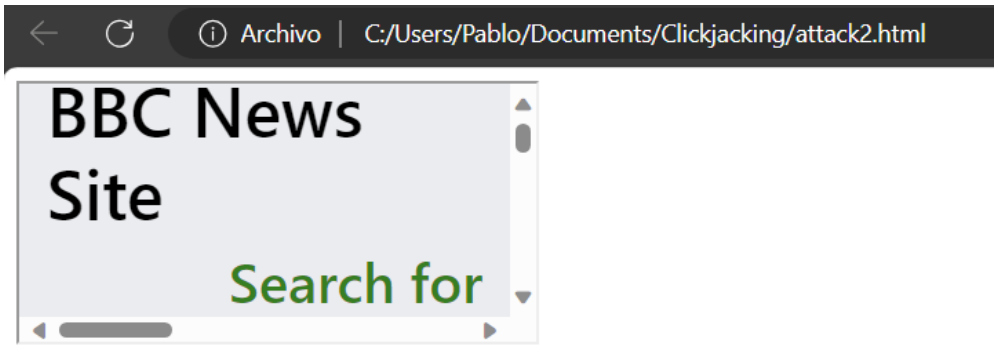


- Dado que el sitio el template jinja2 para generar respuestas HTML de manera dinámica, al utilizar una función insegura que refleja la entrada del usuario en el template, es posible hacer una ejecución remota de código en el servidor.
  Vectores:
  1. Ataques de inyección SSTI
  2. Ejecución remota de código
  3. DoS



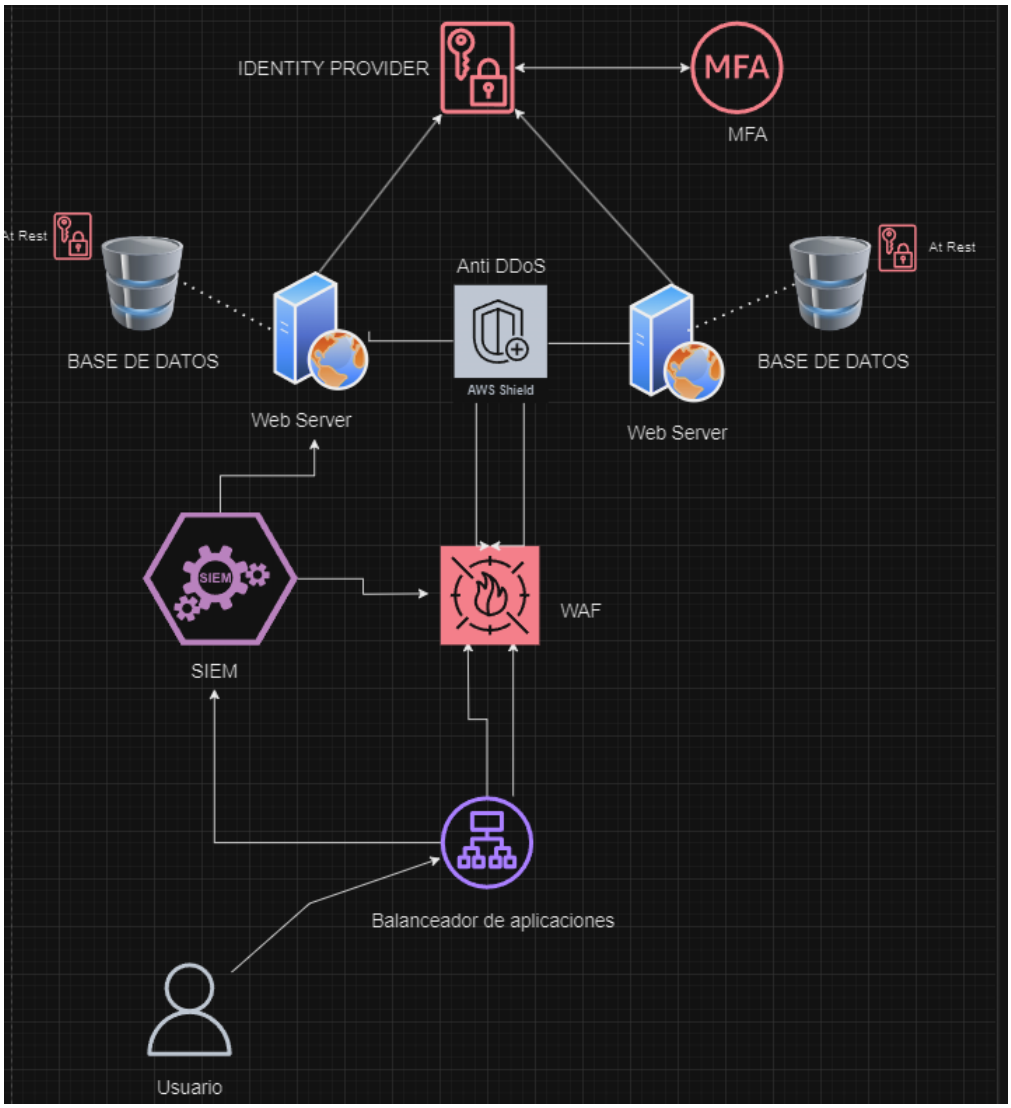


- El sitio carece de medidas para evitar crear frames con el contenido html del sitio lo que lo expone a ataques de tipo clickjacking pues se pueden crear elementos invisibles en el frame para superponerlos en el sitio real.
  Vectores:
  1. Creación de frames transparentes
  2. Ataques de “Watering Hole”
  3. Ataques de descargas “Drive-by”




```
<style>
    iframe {
        position:relative;
        width:$width_value;
        height: $height_value;
        opacity: $opacity;
        z-index: 2;
    }
    div {
        position:absolute;
        top:$top_value;
        left:$side_value;
        z-index: 1;
    }
</style>
<div>Test me</div>
<iframe src="http://127.0.0.1:8000/bbc"></iframe>
```

Propuesta de Arquitectura de seguridad.



Como respuesta a los diferentes hallazgos de seguridad se propone una arquitectura con un esquema de alta disponibilidad con redundancia activa-pasiva con ciertos componentes, características y prácticas a seguir que se explicarán a continuación.

Esquema.

El esquema de alta disponibilidad propone que sean dos servidores alojando la aplicación haciendo uso de dos bases de datos, una que está activa y que alimenta el contenido del sitio y la otra en standby para temas de recuperación de desastres y backups.

Balanceador de carga.

Distribuye el tráfico entre varios servidores, reduciendo la probabilidad de que un solo servidor sea el objetivo de un ataque y contribuye a contener un posible ataque de DoS pues evita la sobrecarga de estos. Adicionalmente, el balanceadora mejora sustancialmente los tiempos de respuesta del sitio por lo que mitiga también algunos ataques que se aprovechan de tiempos de respuesta muy grandes. Puede ofrecer tanto

cifrado como filtrado de tráfico entre cliente y servidor para garantizar información confidencial y además poder descartar a tiempo el tráfico con patrones maliciosos

WAF

Prpoporciona protección contra ataques comunes tipo XSS, CSRF, SSTI, DoS, analiza tráfico cifrado y descifrado con el fin de detectar payloads mas avanzados de ataques y bloquear a tiempo solicitudes. Es uno de los componentes que mayor capa de monitoreo puede brindar.

Anti DDoS

Se propone una solución anti DDoS que específicamente se enfoque en detectar y mitigar ataques que atenten contra la disponibilidad del sitio. Descarta todo el tráfico malicioso y redirige el tráfico legítimo a los servidores. Protege a gran escala los componentes de la arquitectura

Proveedor de Identidades

Es un servidor que permite controlar tanto el proceso de autenticación como de autorización de manera más robusta para acceder a los recursos del sitio web con una capa de protección multifactor por medio de flujos.

MFA

Una vez el proveedor de identidades verifique la autenticidad de las credenciales del usuario, este genera un token de autenticación que es enviado al servidor MFA para que envíe un segundo método ya sea algo que conozca o algo que se tenga.

SIEM

Un sistema estará más protegido en la medida en que esté constantemente monitoreado y auditado. Un SIEM correlaciona eventos de muchas fuentes como balanceador de aplicaciones, Firewalls, IPS, servidor de identidades, logs de los servidores y bases de datos y permite llevar una trazabilidad de los eventos que ocurren en toda la infraestructura.

Mejores prácticas de remediación.

Esta información ya se encuentra consignada en la api que apunta a los endpoints:

- http://localhost:8000/list
  Para listar cada una de las vulnerabilidades encontradas con sus prácticas de remediación a considerar
- http://localhost:8000/vulnid/{id}

Para consultar individualmente las vulnerabilidades encontradas con sus prácticas de remediación a considerar.