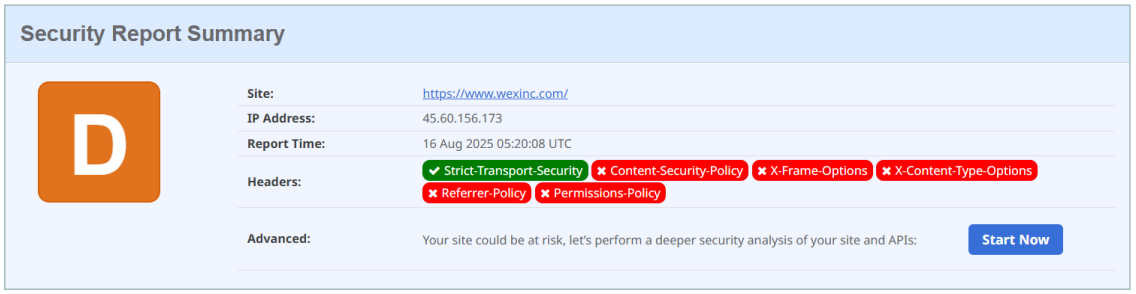
Task 3 - Analyze Website Security Headers Using Online Tools

🔒 Web Security Header Analysis Report

✅ Objective:

Evaluate the presence and effectiveness of key HTTP security headers and assign a **security grade** for each website. For the scan purpose, I used a website called https://securityheaders.com/ from synk.io.

🧪 1. Website: **WEX Inc.** (https://www.wexinc.com)



🔎 Missing Headers:

Header	Present?	Function
X-Frame-Options	❌ No	Prevents **clickjacking** by disallowing the site to be embedded in frames
Content-Security-Policy	❌ No	Defends against **XSS (cross-site scripting)** and **data injection**
Strict-Transport-Security	✅ Yes	Enforces **HTTPS**; prevents **SSL stripping attacks**
X-XSS-Protection	❌ No	Legacy protection against **reflected XSS attacks**

🧾 Grade: D

> Rationale: Only HSTS is set. Lacking most foundational headers, which introduces vulnerabilities to XSS, clickjacking, and content injection.

🧪 2. Website: **Box** (https://www.box.com)

Security Report Summary

A

Site:	https://www.box.com/
IP Address:	103.116.7.21
Report Time:	16 Aug 2025 05:20:20 UTC
Headers:	✔ Content-Security-Policy ✔ Strict-Transport-Security ✔ X-Content-Type-Options ✔ X-Frame-Options ✖ Referrer-Policy ✖ Permissions-Policy
Warning:	Grade capped at A, please see warnings below.
Advanced:	Great grade! Perform a deeper security analysis of your website and APIs: Try Now

🔎 Missing / Weak Headers:

Header	Present?	Notes
X-Frame-Options	✅ Yes	SAMEORIGIN set – good
Content-Security-Policy	⚠️ Yes (Weak)	Contains `'unsafe-inline'`, making it **partially insecure**
Strict-Transport-Security	✅ Yes	Includes `preload`, `includeSubDomains` – excellent
X-XSS-Protection	✅ Yes	Enabled (legacy protection)

🧾 Grade: B+

> Rationale: Strong base, but CSP has 'unsafe-inline', which undermines its protection. Missing Permissions-Policy and Referrer-Policy.

🧪 3. Website: **ChatGPT** (`https://chatgpt.com`)

Security Report Summary

A

Site:	https://chatgpt.com/
IP Address:	104.18.32.47
Report Time:	16 Aug 2025 05:17:03 UTC
Headers:	✔ Content-Security-Policy ✔ Strict-Transport-Security ✔ X-Content-Type-Options ✔ Referrer-Policy ✔ X-Frame-Options ✖ Permissions-Policy
Advanced:	Great grade! Perform a deeper security analysis of your website and APIs: Try Now

🔎 Missing / Weak Headers:

Header	Present?	Notes
X-Frame-Options	❌ No (✔ CSP used instead)	Uses `frame-ancestors` in CSP – **modern, recommended** alternative
Content-Security-Policy	✅ Yes	Strong: Uses nonce and SHA256 hashes for script integrity
Strict-Transport-Security	✅ Yes	Full protection (max-age, preload, subdomains)

Header	Present?	Notes
X-XSS-Protection	❌ No	Omitted (acceptable if CSP is used effectively, which it is)

🧾 Grade: A

> Rationale: Excellent modern setup. CSP replaces older headers. Minor improvements possible (e.g., Permissions-Policy).

Summary of the scan results:

Site	X-Frame-Options	CSP	HSTS	X-XSS-Protection	Grade
WEX Inc.	❌ Missing	❌ Missing	✅ Present	❌ Missing	**D**
Box.com	✅ SAMEORIGIN	⚠️ Weak (`unsafe-inline`)	✅ Present	✅ Present	**B+**
ChatGPT	✅ via CSP	✅ Strong (nonce/sha)	✅ Full (preload)	❌ Missing	**A**

🔐 What Do These Headers Protect Against?

Header	Protection Mechanism
X-Frame-Options	Prevents **clickjacking** by disallowing your page from being framed by other websites
Content-Security-Policy (CSP)	Defends against **XSS, data injection, mixed content**, and restricts asset loading sources
Strict-Transport-Security (HSTS)	Forces **HTTPS**, blocks **SSL downgrade attacks**, and ensures data is encrypted in transit
X-XSS-Protection	Legacy browser feature that blocks **reflected XSS attacks**

🛠️ How Can These Sites Improve?

WEX Inc.

- ✅ **Top Priority**: Implement missing headers. This is a **high-risk** configuration.
- 🔧 **Suggested Headers:**

```
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
Content-Security-Policy: default-src 'self'; object-src 'none'; frame-ancestors 'self';
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
```

```
Referrer-Policy: strict-origin-when-cross-origin
Permissions-Policy: geolocation=(), microphone=(), camera=()
```

Box

- ✅ Harden CSP by **removing** `'unsafe-inline'` in `default-src`
- ✅ Add `Referrer-Policy` and `Permissions-Policy`

ChatGPT

- ✅ Already excellent. Just add `Permissions-Policy` if not defined in full
- ❌ Avoid `unsafe-inline` in style if present

Learning Outcomes & Key Concepts

✔️ Understanding Security Headers

- Headers work as **browser-enforced security policies**
- Prevent common web threats: **XSS, clickjacking, insecure content loading**

Interview Questions:

1. What are HTTP security headers?

Answer:

HTTP security headers are directives included in HTTP responses that instruct the browser how to handle content. They help protect web applications against common threats like cross-site scripting (XSS), clickjacking, and protocol downgrade attacks by enforcing security-related policies at the browser level.

---

2. Name five common security headers.

Answer:

- Content-Security-Policy
- X-Frame-Options
- Strict-Transport-Security
- X-Content-Type-Options
- Referrer-Policy

---

3. What does `X-Content-Type-Options: nosniff` do?

Answer:

It prevents browsers from MIME type sniffing a response away from the declared `Content-Type`. This reduces the risk of executing malicious files that may have misleading MIME types.

---

4. Why is `Strict-Transport-Security` important?

Answer:

It ensures that browsers only communicate with the site over HTTPS by enforcing secure connections and preventing protocol downgrade attacks. This header is crucial for maintaining the confidentiality and integrity of data in transit.

---

5. What's the purpose of `Content-Security-Policy`?

Answer:

CSP restricts the sources from which content like JavaScript, CSS, images, and other resources can be loaded. It helps prevent cross-site scripting (XSS), code injection, and data exfiltration attacks.

---

6. How does `X-Frame-Options` prevent clickjacking?

Answer:

It prevents a web page from being embedded in a frame or iframe from another origin. This defends against clickjacking, where a user is tricked into clicking something different from what they perceive.

---

7. Can security headers replace web application firewalls (WAFs)? Why or why not?

Answer:

No. Security headers provide client-side protection by instructing browsers, while WAFs provide server-side protection by filtering, monitoring, and blocking malicious traffic. Both are necessary components of a defense-in-depth strategy.

---

8. What is the difference between `X-XSS-Protection` and `Content-Security-Policy`?

Answer:

`X-XSS-Protection` is a legacy header that activates a browser's built-in XSS filter. It's largely obsolete in modern browsers.

`Content-Security-Policy` is a modern, flexible header that can prevent XSS and other attacks by controlling the sources of executable content.

---

9. What grade did the scanned website get, and what was missing?

Answer:

The WEX Inc. website received a grade of D. It was missing the following security headers:

- Content-Security-Policy
- X-Frame-Options
- X-Content-Type-Options

- Referrer-Policy
- Permissions-Policy

  It only had `Strict-Transport-Security` implemented.

---

10. What steps would you recommend to improve header security?

Answer:

- Implement `Content-Security-Policy` with restrictive source definitions
- Add `X-Frame-Options: SAMEORIGIN` to prevent framing
- Add `X-Content-Type-Options: nosniff` to prevent MIME sniffing
- Include `Referrer-Policy` and `Permissions-Policy` for added privacy and control
- Ensure `Strict-Transport-Security` includes `preload` and `includeSubDomains`
- Regularly audit and test headers using tools like securityheaders.com or Mozilla Observatory

---