SAMPLE WEB SECURITY REVIEW REPORT

Demo Client: Example Startup Website

Prepared by Alekh Verma - Cybersecurity Intern / Web Security Learner

Fortinet Certified Associate Cybersecurity (FCA) | Cisco NetAcad | Web Security | OSINT

Report Type	Sample / Portfolio Demonstration
Assessment Date	June 28, 2026
Report ID	AVR-DEMO-WEB-2026-001
Scope	Authorized demo web security review only

IMPORTANT NOTICE

Use this report only as a portfolio sample. It is not a real audit of any public website.

Prepared as a professional proof-of-work sample by Alekh Verma - Fortinet FCA Certified Cybersecurity Practitioner focused on web security, OSINT, FortiGate, secure systems, and client-safe security documentation.

Legal and ethical boundary

All security testing must be performed only on systems you own or where you have clear written permission. This sample intentionally avoids exploit steps, harmful instructions, or unsafe testing guidance.

EXECUTIVE SUMMARY

This sample report demonstrates how a professional, client-safe web security review can be communicated to clients or recruiters. The fictional assessment reviewed a demo startup website for common web security hygiene issues, public exposure risks, configuration weaknesses, and documentation quality. The overall sample risk posture is assessed as MEDIUM because multiple issues could increase business risk if left unresolved, but no active exploitation or unauthorized testing was performed.

Authorized scope only	No illegal testing	Client-safe language	Actionable remediation

Key outcomes:

- Six sample findings were documented across security headers, form abuse protection, dependency hygiene, public repository exposure, error handling, and privacy transparency.
- Most fixes are low-to-medium effort and can meaningfully improve the website security posture.
- The report focuses on business impact, practical recommendations, and responsible communication instead of attack instructions.

SCOPE AND RULES OF ENGAGEMENT

Item	Details
Target	demo.example-startup.local (fictional demonstration target)
Testing type	Passive review, configuration review, client-provided screenshots, and documentation checklist
Included	Homepage, contact form, public assets, security headers, public repository hygiene, privacy page, and dependency review
Excluded	Credential attacks, phishing, malware, account takeover, denial-of-service, real exploitation, or testing outside written scope
Assumption	Client has ownership/authorization for all reviewed assets

METHODOLOGY

- Reviewed visible website pages and public metadata for security hygiene and privacy clarity.
- Checked common defensive controls such as security headers, HTTPS posture, cookie flags, and basic error behavior.
- Reviewed exposed public files and repository hygiene signals for accidental secrets or sensitive information patterns.
- Assessed contact form abuse risk from a defensive perspective such as rate limiting, validation, and spam protection.
- Mapped observations to simple risk levels: Critical, High, Medium, Low, and Informational.

RISK RATING GUIDE

Rating	Meaning	Example Business Impact
Critical	Immediate risk with severe business impact	Data breach, unauthorized access, major outage
High	Important weakness requiring priority remediation	Sensitive data exposure or account security risk
Medium	Practical issue that increases attack surface	Abuse, misconfiguration, preventable exposure
Low	Security hygiene improvement	Reduced trust, weaker hardening, compliance gaps
Info	Observation or improvement note	Documentation, monitoring, or maturity improvement

FINDINGS OVERVIEW

A dashboard-style summary of the sample observations before detailed findings.

ID	Finding	Risk	Status
WEB-01	Missing or incomplete security headers	Medium	Open
WEB-02	Contact form lacks visible abuse protection	Medium	Open
WEB-03	Public repository may expose sensitive project context	High	Open
WEB-04	Dependency review process is not documented	Medium	Open
WEB-05	Error handling messages may reveal internal details	Low	Open
WEB-06	Privacy and data-use notice needs clearer language	Low	Open

Top Remediation Priorities

Priority	Focus Area	Why it matters
1	Repository and secret hygiene	Prevents accidental exposure of tokens, keys, and sensitive deployment details.
2	Browser hardening and headers	Improves protection against common web security and trust issues.
3	Contact form abuse controls	Reduces spam, automated abuse, and operational noise for small teams.

Sample maturity note

This report is written to show clear client communication: risk, impact, recommendation, and closure criteria. It avoids unsafe exploit instructions and stays within defensive security documentation.

WEB-01 - MISSING OR INCOMPLETE SECURITY HEADERS

Field	Value
Risk	Medium
Affected Area	Demo web application / public website configuration
Status	Open - sample recommendation

Risk Level: Medium

This is a sample risk rating intended for client-friendly reporting and training.

Observation

Security headers help browsers enforce safer behavior. The demo website appears to rely mostly on default browser behavior instead of explicitly defining protective headers.

Business Impact

Without clear browser-side hardening, users may be more exposed to clickjacking, mixed content problems, overly broad resource loading, or weaker transport security expectations.

Recommended Remediation

- Define a strict Content Security Policy appropriate to the site.
- Use HSTS after confirming all subdomains support HTTPS.
- Set frame restrictions using frame-ancestors or an equivalent policy.
- Review Referrer-Policy and Permissions-Policy based on business needs.

Validation / Closure Criteria

- Client or developer confirms the recommendation has been applied.
- A follow-up review verifies the issue no longer appears in the approved scope.

WEB-02 - CONTACT FORM LACKS VISIBLE ABUSE PROTECTION

Field	Value
Risk	Medium
Affected Area	Demo web application / public website configuration
Status	Open - sample recommendation

Risk Level: Medium

This is a sample risk rating intended for client-friendly reporting and training.

Observation

The contact workflow does not clearly show rate limiting, bot protection, validation strategy, or abuse monitoring.

Business Impact

Automated spam or high-volume abuse could overload inboxes, damage reputation, or distract from real client messages.

Recommended Remediation

- Add server-side validation and rate limiting.
- Use a lightweight anti-spam control appropriate for user experience.
- Log form submissions safely without storing unnecessary personal data.
- Add clear success/failure messages without exposing internal behavior.

Validation / Closure Criteria

- Client or developer confirms the recommendation has been applied.
- A follow-up review verifies the issue no longer appears in the approved scope.

WEB-03 - PUBLIC REPOSITORY MAY EXPOSE SENSITIVE PROJECT CONTEXT

Field	Value
Risk	High
Affected Area	Demo web application / public website configuration
Status	Open - sample recommendation

Risk Level: High

This is a sample risk rating intended for client-friendly reporting and training.

Observation

Public repositories are useful for portfolio proof, but they should be reviewed for accidental exposure of secrets, API keys, tokens, private notes, or deployment metadata.

Business Impact

A leaked token or configuration file could allow unauthorized access to third-party services or reveal internal architecture.

Recommended Remediation

- Run a secret scanning tool before every public push.
- Keep .env files and deployment secrets out of the repository.
- Review commit history for accidental secret exposure.
- Rotate any exposed token immediately if a leak is discovered.

Validation / Closure Criteria

- Client or developer confirms the recommendation has been applied.
- A follow-up review verifies the issue no longer appears in the approved scope.

WEB-04 - DEPENDENCY REVIEW PROCESS IS NOT DOCUMENTED

Field	Value
Risk	Medium
Affected Area	Demo web application / public website configuration
Status	Open - sample recommendation

Risk Level: Medium

This is a sample risk rating intended for client-friendly reporting and training.

Observation

The demo website uses third-party libraries. There is no visible process for reviewing outdated packages, known vulnerabilities, or dependency changes.

Business Impact

Known vulnerable dependencies may remain in production longer than necessary and create avoidable business risk.

Recommended Remediation

- Use dependency auditing as part of the release process.
- Pin and review major dependency updates before production release.
- Maintain a simple dependency update log.
- Remove unused packages to reduce attack surface.

Validation / Closure Criteria

- Client or developer confirms the recommendation has been applied.
- A follow-up review verifies the issue no longer appears in the approved scope.

WEB-05 - ERROR HANDLING MESSAGES MAY REVEAL INTERNAL DETAILS

Field	Value
Risk	Low
Affected Area	Demo web application / public website configuration
Status	Open - sample recommendation

Risk Level: Low

This is a sample risk rating intended for client-friendly reporting and training.

Observation

Some application states may show technical messages that are useful for debugging but not ideal for public users.

Business Impact

Verbose errors can reveal implementation details that help attackers understand internal technology choices.

Recommended Remediation

- Use generic user-facing error messages.
- Send detailed errors only to private logs.
- Review 404, 500, form, and API error pages.
- Add a simple incident triage process for unexpected errors.

Validation / Closure Criteria

- Client or developer confirms the recommendation has been applied.
- A follow-up review verifies the issue no longer appears in the approved scope.

WEB-06 - PRIVACY AND DATA-USE NOTICE NEEDS CLEARER LANGUAGE

Field	Value
Risk	Low
Affected Area	Demo web application / public website configuration
Status	Open - sample recommendation

Risk Level: Low

This is a sample risk rating intended for client-friendly reporting and training.

Observation

A website that collects contact form submissions should clearly explain what data is collected, why it is collected, and how long it is retained.

Business Impact

Unclear data handling can reduce client trust and create avoidable compliance questions.

Recommended Remediation

- Add a short privacy note near the contact form.
- Explain what information is collected and why.
- Avoid collecting unnecessary personal data.
- Define a reasonable retention period for inquiry messages.

Validation / Closure Criteria

- Client or developer confirms the recommendation has been applied.
- A follow-up review verifies the issue no longer appears in the approved scope.

REMEDIATION ROADMAP

Priority	Action	Owner	Suggested Timeframe
P1	Review public repository for secrets and rotate anything exposed	Developer / Owner	Same day
P2	Add missing security headers and test site behavior	Developer	1-3 days
P2	Add contact form validation and abuse protection	Developer	1-3 days
P3	Document dependency review workflow	Developer / Maintainer	1 week
P3	Improve public privacy notice and retention language	Owner	1 week
P3	Clean up public error messages and logging workflow	Developer	1 week

CLIENT-SAFE SUMMARY

Overall conclusion

The demo website shows a good base, but several security hygiene improvements are recommended before treating it as production-ready. The highest priority is public repository hygiene, followed by browser hardening, form abuse controls, dependency review, and clearer privacy communication.

PORTFOLIO USE STATEMENT

- This sample report may be used to demonstrate report writing, security communication, and structured web security review skills.
- The report does not claim unauthorized testing or exploitation of any real target.
- All examples are fictional and written for ethical, legal, and defensive cybersecurity learning.

APPENDIX A - WEBSITE SECURITY CHECKLIST

Control	Status	Notes
HTTPS enforced	To verify	Confirm redirects and certificate configuration
HSTS	Recommended	Enable only after confirming HTTPS coverage
Content Security Policy	Recommended	Start with report-only if needed
Frame protection	Recommended	Use frame-ancestors or equivalent
Form rate limiting	Recommended	Prevent spam and automated abuse
Secrets scan	High priority	Check repo and deployment history
Dependency audit	Recommended	Add to release checklist
Privacy notice	Recommended	Explain contact data handling

APPENDIX B - REPORT QUALITY PRINCIPLES

- Be clear: write findings so a non-technical client can understand the business risk.
- Be ethical: never test systems without clear permission and scope.
- Be practical: every finding should include a realistic remediation path.
- Be honest: never exaggerate experience, risk, or impact.
- Be safe: do not include harmful proof-of-concept steps in public portfolio reports.

PREPARED BY

Name	Profile
Alekh Verma	Cybersecurity learner focused on web application security, OSINT, secure systems, and responsible security documentation.
Portfolio	https://alekhverma.vercel.app/
LinkedIn	www.linkedin.com/in/alekhverma
GitHub	https://github.com/officialalekh
Credential	Fortinet Certified Associate Cybersecurity (FCA)

Final note

This document is a portfolio-ready sample. It should be customized before use with real clients, and real testing should only happen after written authorization and defined scope.