

WEBSITE SECURITY REPORT

example.com

About Written by hand for the reserved address example.com. Not a scan of a real website.

Findings We found 8 items to review: 2 medium, 4 low, 2 informational.



CONTENTS

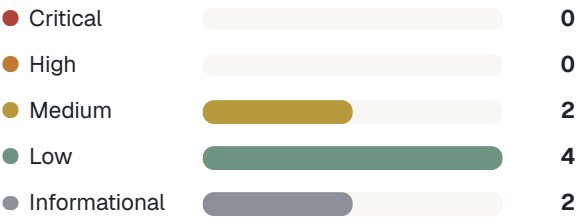
Executive summary	2
Findings overview: your checklist	3
What we found and how to fix it	4
Appendix: scope, health by area, evidence, technologies, data sources	8

EXECUTIVE SUMMARY

Your website at a glance

We found 8 items to review: 2 medium, 4 low, 2 informational.

Findings by severity



Fix first

#	ISSUE AND WHY IT MATTERS	SEVERITY	EFFORT	PAGE
1	Keep the login cookie on HTTPS	Medium	Minutes	p. 4
2	Update an outdated component	Medium	Hours	p. 4
3	Tell browsers to always use HTTPS	Low	Minutes	p. 5
5	Publish a way to report a security problem	Low	Minutes	p. 5
6	Add a DMARC record for your domain	Low	Minutes	p. 6

What we checked

Your website

How visitors connect, and the protections their browser switches on.

5 findings

Your email and domain

Whether others can send email as you, and how your domain records are set up.

2 findings

Your software and files

Old versions your site gives away, and keys or files left where anyone can read them.

1 finding

Your reputation

Whether Google lists your site as unsafe, so you hear before your visitors are warned.

No findings

Send this to your developer. Each finding says where we saw it, what we expected to see, and how to fix it, step by step, with a quick way to check the fix worked.

FINDINGS OVERVIEW

Your checklist

Every finding, most serious first. Work down the list and tick each one off once its check passes.

#	ISSUE	SEVERITY	AREA	EFFORT	PAGE	☐
1	Keep the login cookie on HTTPS	Medium	Web	Minutes	p. 4	☐
2	Update an outdated component	Medium	Software	Hours	p. 4	☐
3	Tell browsers to always use HTTPS	Low	Web	Minutes	p. 5	☐
4	Add a Content Security Policy	Low	Web	Days	p. 5	☐
5	Publish a way to report a security problem	Low	Web	Minutes	p. 5	☐
6	Add a DMARC record for your domain	Low	Email security	Minutes	p. 6	☐
7	Your site answers over HTTPS and redirects plain HTTP	Informational	Web	Minutes	p. 6	☐
8	SPF record found	Informational	Email security	Minutes	p. 6	☐

FINDINGS

What we found and how to fix it

After a fix, run its check, then run a re-check from your dashboard to update this report.

MEDIUM · 2

#1 Keep the login cookie on HTTPS

WebEffort: minutesConfidence: confirmed

Why it matters: The session cookie is set without the Secure flag. Adding it keeps the cookie on encrypted connections, so signed-in sessions stay private, even on public Wi-Fi.

Where: example.com

WHAT WE SAW

We saw (1): Set-Cookie: PHPSESSID=...; path=/

HOW TO FIX

1. Add the Secure and HttpOnly flags (and SameSite=Lax) to session cookies.
2. In PHP: session.cookie_secure = 1 and session.cookie_httponly = 1.

KEY EVIDENCE

We saw (2)	↑ no Secure, no HttpOnly
A fixed site shows	Set-Cookie: PHPSESSID=...; path=/; Secure; HttpOnly; SameSite=Lax

Full evidence: appendix C

#2 Update an outdated component

SoftwareEffort: hoursConfidence: likely

Why it matters: Your homepage announces PHP 7.4. Moving to a supported version brings back regular security updates and closes the issues public vulnerability data lists for 7.4.

Where: example.com

WHAT WE SAW

We saw (1): X-Powered-By: PHP/7.4.3

HOW TO FIX

1. Ask your host to move the site to a supported PHP version (8.x).
2. Then stop advertising the version at all: set expose_php = Off.

KEY EVIDENCE

We saw (2)	PHP 7.4 — end of life 2022-11-28
A fixed site shows	A supported PHP 8 release (8.3 or later), version not announced (expose_php = Off)

Full evidence: appendix C

● LOW · 4

#3 Tell browsers to always use HTTPS

Low

Web

Effort: minutes

Confidence: confirmed

Why it matters: An HSTS header sends browsers straight to https://, even when a visitor types your address without it, so after the first visit, every return visit goes straight to an encrypted connection. Adding your site to the HSTS preload list covers the very first visit too.

Where: example.com

WHAT WE SAW

We saw: Strict-Transport-Security: (header absent)

HOW TO FIX

1. Send Strict-Transport-Security: max-age=31536000; includeSubDomains once every subdomain works over HTTPS.

KEY EVIDENCE

A fixed site shows

Strict-Transport-Security: max-age=31536000; includeSubDomains

Full evidence: appendix C

#4 Add a Content Security Policy

Low

Web

Effort: days

Confidence: confirmed

Why it matters: A Content Security Policy tells browsers which scripts your pages may load. A restrictive, well-configured CSP can limit what injected script loads or where it sends data.

Where: example.com

WHAT WE SAW

We saw: Content-Security-Policy: (header absent)

HOW TO FIX

1. Start with Content-Security-Policy-Report-Only to learn what the site legitimately loads, then enforce the policy.

Full evidence: appendix C

#5 Publish a way to report a security problem

Low

Web

Effort: minutes

Confidence: confirmed

Why it matters: A security.txt file gives anyone who spots a problem with your site a clear, safe way to tell you.

Where: example.com

WHAT WE SAW

We saw: GET /.well-known/security.txt → 404

HOW TO FIX

1. Publish /.well-known/security.txt with a Contact: line and an Expires: date.

Full evidence: appendix C

#6 Add a DMARC record for your domain

Low

Email security

Effort: minutes

Confidence: confirmed

Why it matters: A DMARC record tells receiving mail servers what to do with email that claims to come from your domain, so they can turn away mail that only pretends to be you.

Where: example.com

WHAT WE SAW

We saw: `_dmarc.example.com` TXT (no record)

HOW TO FIX

1. Publish a `_dmarc` TXT record.
2. Start with `v=DMARC1; p=none; rua=mailto:dmarc@your-domain` to monitor, then move to `p=quarantine`.

KEY EVIDENCE

A fixed site shows `_dmarc.example.com` TXT "v=DMARC1; p=none; rua=mailto:dmarc@example.com"

Full evidence: appendix C

● INFORMATIONAL · 2

#7 Your site answers over HTTPS and redirects plain HTTP

Informational

Web

Effort: minutes

Confidence: confirmed

Why it matters: Visitors who arrive over `http://` are moved to an encrypted connection.

Where: example.com

WHAT WE SAW

We saw (1): GET `http://example.com/` → 301

HOW TO FIX

1. Visitors who arrive over `http://` are moved to an encrypted connection.

Nothing to do.

KEY EVIDENCE

We saw (2) GET `https://example.com/` → 200

Full evidence: appendix C

#8 SPF record found

Informational

Email security

Effort: minutes

Confidence: confirmed

Why it matters: Receiving mail servers can check which servers are allowed to send email for your domain.

Where: example.com

WHAT WE SAW

We saw: `example.com` TXT `v=spf1 include:_spf.mail.example -all`

HOW TO FIX

1. Receiving mail servers can check which servers are allowed to send email for your domain.

Nothing to do. Pair it with DMARC so the policy is enforced.

Full evidence: [appendix C](#)

APPENDIX

The details, for you or your developer

A Scope and method

A Quick Scan looks at your website from the outside, the way a visitor's browser does, at the moment we check. We don't log in, fill in forms or test pages behind a sign-in.

Your website: HTTPS and redirects, Encryption (TLS) test, Security headers, Cookie security

Your email and domain: Email security (SPF, DMARC, DKIM), DNS health

Your software and files: Outdated software and plugins, Known vulnerabilities, Exposed keys, Exposed files (verified sites)

Your reputation: Google blocklist

Website	https://example.com/
Checks	11 ran
Server software	Server: nginx · X-Powered-By: PHP/7.4.3
HTTPS	HTTPS works · HTTP redirects to HTTPS · HSTS off
Email	SPF set up · no DMARC

C Evidence for each finding

#1 Keep the login cookie on HTTPS

Where	example.com
We saw (1)	Set-Cookie: PHPSESSID=...; path=/
We saw (2)	↑ no Secure, no HttpOnly
A fixed site shows	Set-Cookie: PHPSESSID=...; path=/; Secure; HttpOnly; SameSite=Lax
Confidence and references	Confidence: Confirmed

#2 Update an outdated component

Where	example.com
We saw (1)	X-Powered-By: PHP/7.4.3
We saw (2)	PHP 7.4 — end of life 2022-11-28
A fixed site shows	A supported PHP 8 release (8.3 or later), version not announced (expose_php = Off)
Confidence and references	Confidence: Likely

#3 Tell browsers to always use HTTPS

Where	example.com
We saw	Strict-Transport-Security: (header absent)
A fixed site shows	Strict-Transport-Security: max-age=31536000; includeSubDomains
Confidence and references	Confidence: Confirmed

#4 Add a Content Security Policy

Where	example.com
We saw	Content-Security-Policy: (header absent)
Confidence and references	Confidence: Confirmed

#5 Publish a way to report a security problem

Where	example.com
We saw	GET /.well-known/security.txt → 404
Confidence and references	Confidence: Confirmed

#6 Add a DMARC record for your domain

Where	example.com
We saw	_dmarc.example.com TXT (no record)
A fixed site shows	_dmarc.example.com TXT "v=DMARC1; p=none; rua=mailto:dmarc@example.com"
Confidence and references	Confidence: Confirmed

#7 Your site answers over HTTPS and redirects plain HTTP

Where	example.com
We saw (1)	GET http://example.com/ → 301
We saw (2)	GET https://example.com/ → 200
Confidence and references	Confidence: Confirmed

#8 SPF record found

Where	example.com
We saw	example.com TXT v=spf1 include:_spf.mail.example -all
Confidence and references	Confidence: Confirmed

D Technologies we found

JavaScript libraries

TECHNOLOGY	VERSION	STATUS	HOW WE KNOW
jQuery	3.6.0	No known issues	Script URL on the homepage. No known issues matched

Server software

TECHNOLOGY	VERSION	STATUS	HOW WE KNOW
PHP	7.4.3	End of life (end of life since 2022-11-28)	X-Powered-By header. End of life
nginx	Not shown	Not checked for known issues	Server header. Version not announced

E Data sources

- Vulnerability and exploit data in this report comes from third parties:
- NVD (National Vulnerability Database, NIST): "This product uses the NVD API but is not endorsed or certified by the NVD." nvd.nist.gov
 - CISA Known Exploited Vulnerabilities Catalog (public domain). Not endorsed by CISA or DHS. cisa.gov/known-exploited-vulnerabilities-catalog
 - EPSS (Exploit Prediction Scoring System) by FIRST.org. first.org/epss
 - OSV.dev, including the GitHub Advisory Database (CC BY 4.0) and other upstream databases under their own licences. osv.dev · [OSV data sources and licences](#)
 - Wordfence Intelligence. Copyright Defiant Inc.; used under the Wordfence Intelligence licence. Each WordPress vulnerability links to its record at wordfence.com. wordfence.com/threat-intel
 - Support dates from endoflife.date. endoflife.date
 - JavaScript library detection patterns from Retire.js (Apache-2.0). retirejs.github.io
- ScanDefend is not affiliated with or endorsed by any of these organisations.