FREE WEBSITE SECURITY CHECK

What your website scan found

The complete guide

How to read your grade. What each finding means. Who fixes it. What to do this week, this month and this quarter.

INSIDE

16

kinds of problem, each in plain words. Each one says who fixes it and how long it takes.

3

time frames for your fix plan. You also get a worksheet and a checklist to print.

4

emails to copy. Send them to your developer, a site owner, an insurer or your boss.

Also: what a scan can't see, what a clean result means, testing with Argos Phantom, and word list.

Intelligent Automation, LLC | (888) 711-4521 | sales@intelamation.com

Edition: November 2026

THE COMPLETE GUIDE

Contents

01 How to read this guide 3

02 If you only have 10 minutes 4

03 What the scan is, and what it is not 5

04 Why attackers start on the outside 7

05 Reading your result 8

06 The 16 kinds of problem, in plain words 11

07 What a scan can't see, and what a penetration test finds 16

08 Your fix plan 17

09 Emails you can copy 19

10 If your result is clean 20

11 When you need a real penetration test 22

12 Your next steps with IA 24

13 Questions people ask 27

14 Words you will see (glossary) 28

Two names for the same two checks

The website calls them the **free check** and the **deeper check**. The PDF report calls them the **open check** and the **verified check**. This guide uses the website's names.

This guide explains the Website Security Check as it works in November 2026. The checks, the limits and the words in each finding come from the scanner itself. You can get the newest copy at intelamation.com/Website-Security-Scan-Guide.pdf. The guide is free help to get you ready. It is not a security audit. It is not a penetration test. It is not legal, insurance or compliance advice. A good result means we found fewer known problems in what we checked, on that day. It does not mean your website, your business or your data are safe. All times in this guide are typical guesses for planning. They are not promises. Trademarks belong to their owners.

SECTION 1

How to read this guide

You don't need to be technical to use this guide. It is written for the person who owns or runs the business.

> **In plain words**
>
> The scan looked at your website from the outside. This guide tells you what it found, what each thing means and who should fix it.

How the guide is built

- **Plain words first.** We use everyday words. When a technical word matters, we explain it once and then keep using the plain word.
- **"In plain words" boxes.** Every big section and every kind of problem starts with a short box like the one above. Read the boxes and you will have the main idea.
- **The report's own words in bold.** When we quote a finding exactly as your report shows it, we put it in bold. That way you can match it to your report.
- **Typical times.** When we say how long a fix takes, it is a rough guess to help you plan. It is not a promise.
- **Words you will see.** Section 14 is a glossary. It explains every technical word in this guide.

Who should read which part

If you are	Read
The owner, short on time	Section 2, then the fix plan in section 8.
The person who will chase the fixes	Sections 5, 6 and 8, and the worksheet at the back.
Your web developer or host	The report itself, section 6 and the email in section 9.
Someone whose result was clean	Section 10.
Someone who was asked for a penetration test	Sections 7 and 11.

SECTION 2

If you only have 10 minutes

Read these five things. They cover what matters most.

In plain words

Find your worst problems. Give each one to the right person with a date. Check again when they say it's fixed.

1	**Your grade** See how the score works.	Section 5
2	**Your worst findings** Look up each one. See who fixes it.	Section 6
3	**Your fix plan** This week, this month, this quarter.	Section 8
4	**The email to your developer** Copy email A and send it.	Section 9
5	**When to scan again** Once a month, and after changes.	Section 10

The 3 things to do today

1. Look at your grade and the findings marked critical or high (section 5).
2. Send your report and the email in section 9 to whoever looks after your website.
3. Put a date in your calendar to run the scan again after they fix things.

Want help instead?

Book a free 45-minute call with us at meetings.intelamation.net/schedule. You can also call (888) 711-4521. We will go through your result with you. There is no obligation.

SECTION 3

What the scan is, and what it is not

The scan looks at your website the way anyone on the internet can. It does not log in. It needs nothing installed. It never touches your office network.

> **In plain words**
>
> Think of someone walking around the outside of your shop. They check the doors and windows. They don't go inside.

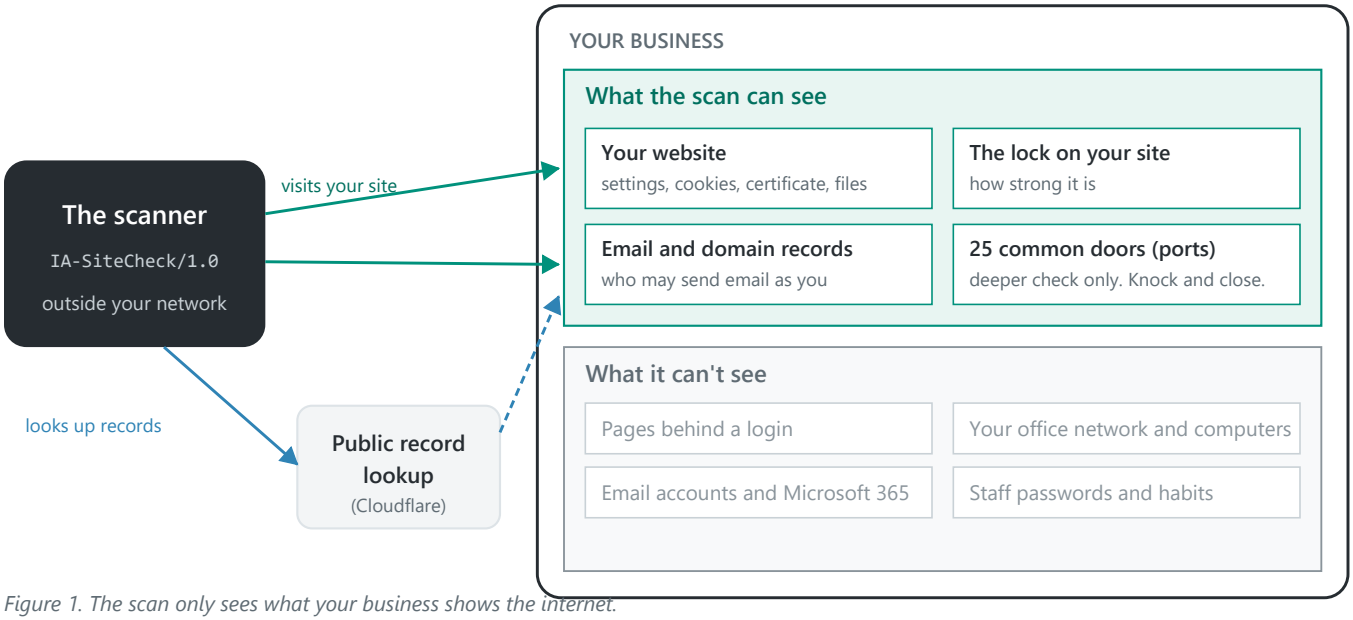


Figure 1. The scan only sees what your business shows the internet.

The scan gets everything from three places. First, the answers your website gives to anyone who visits. Second, your domain's public records. These are like the internet's address book, and they also hold your email settings. Third, in the deeper check, a short list of "doors" on your server. We call these doors ports. The scan looks up your records through a public service run by Cloudflare. It does not ask your own servers for them.

The two levels

	Free check	Deeper check
Who can run it	Anyone, after a quick check that you are a person.	Only someone who proves they control the domain. They must also confirm they may test it.
What it sends	About 15 requests. It loads your home page and your plain http address. It asks for 10 files that should never be public. It asks for your security.txt file. It makes up to 3 test connections to check the lock on your site.	The free check again. Then requests that only read. It sends 1 OPTIONS request and 1 TRACE request to your home page. It also looks up records and knocks on a short list of doors. It sends at most 4 a second and 200 in total.
Time limit	60 seconds.	110 seconds for the deeper part. About 3 minutes for the whole job.

What it covers	The lock on your site and the settings it sends to browsers. Cookies. Email and domain records. Private files. The software your site names.	All of that. Plus login pages, risky request types and links that bounce visitors. Plus page inputs, old software, old web addresses and 25 common doors.
What you get	A grade and a list of findings on the results page.	The same results page. When email is switched on, we also send a PDF report to the work email you used.

What the scan never does

- It never tries a password. It never logs in. It never looks behind a login.
- It never sends a form that uses POST. It never sends PUT or DELETE. It never sends a request body.
- It skips links and forms that look like actions, such as delete, logout, unsubscribe or confirm.
- It never breaks in. It looks for signs of a weakness and stops there.
- It never connects to private network addresses. It refuses domains on its blocked list.

How to prove you own the domain

The deeper check asks for a work email at your domain. It also asks for one of three proofs. Any one is enough. But they unlock different amounts.

- **A small file or a tag.** You put a small file at `/.well-known/ia-site-check.txt` with the code we give you. Or you add a tag to the top of your home page. Either one unlocks the whole deeper check. Every address your domain points to must show the same proof. We check up to 4 addresses.
- **A domain record.** You add a text record at `_ia-site-check.yourdomain`. This proves you control your domain's records. It does not prove you control the servers. So it only unlocks the check of your old web addresses.

If you don't prove it within 1 hour, the request runs out. Once you prove it, it lasts until 24 hours after you started.

> **How to spot the scan in your logs**
>
> Every request we send says `IA-SiteCheck/1.0`. The deeper check also asks for 1 made-up page. Its name starts with `/ia-site-check-`. This shows us what your "not found" page looks like. It looks up 1 made-up web address that starts with `ia-sc-`. It uses the address `canary.invalid` to test links that bounce visitors. It sends a test word that starts with `iasc` to page inputs. None of these can change anything on your site.

What it is not

The scan is not a penetration test. A penetration test is a planned attempt to break in. Qualified people run it, with your written permission. A test finds things a scan cannot. It looks at how logins, payments and customer records work. The scan also does not look at your email accounts or your Microsoft 365 settings. It does not look at your staff's computers or your office. Treat the scan as a fast, honest first look from the outside. Section 11 explains when you need a test.

> **Fair-use limits**
>
> Each visitor can run 6 free checks an hour. Each visitor can run 3 deeper checks an hour. If you scan the same domain again within 10 minutes, you get the earlier result back. We keep results for about 15 minutes. Then we delete them.

SECTION 4

Why attackers start on the outside

Anyone can reach your website and your email domain. That is where most attacks begin. It is also the one place you can check today without installing anything.

> **In plain words**
>
> Your website is your shop window. Anyone can walk past it. So it's the first place to check.

Most security money goes on the inside. It goes on computers, virus protection, email filters and backups. The outside often gets nobody's attention. A website is built once. It is handed to a host and left alone. A door on a server is opened for a vendor and never closed. A test copy of the site goes up for a launch and is forgotten. None of these show up on an office checklist. They don't live in the office.

An attacker doesn't need a reason to look at you. Computer programs work through web addresses all day. They write down what answers. If something useful answers, a person takes a closer look. So ask yourself one question. What does my business show the internet? Would I be happy if a stranger looked closely at it?

What sits on your outside

Your "outside" is everything a stranger can reach from the internet without your permission. Security people call this your attack surface. For most small businesses it falls into five groups. The free and deeper checks cover your website, your email and domain records, common doors and old web addresses. Cloud accounts and user logins need access you give a tester.

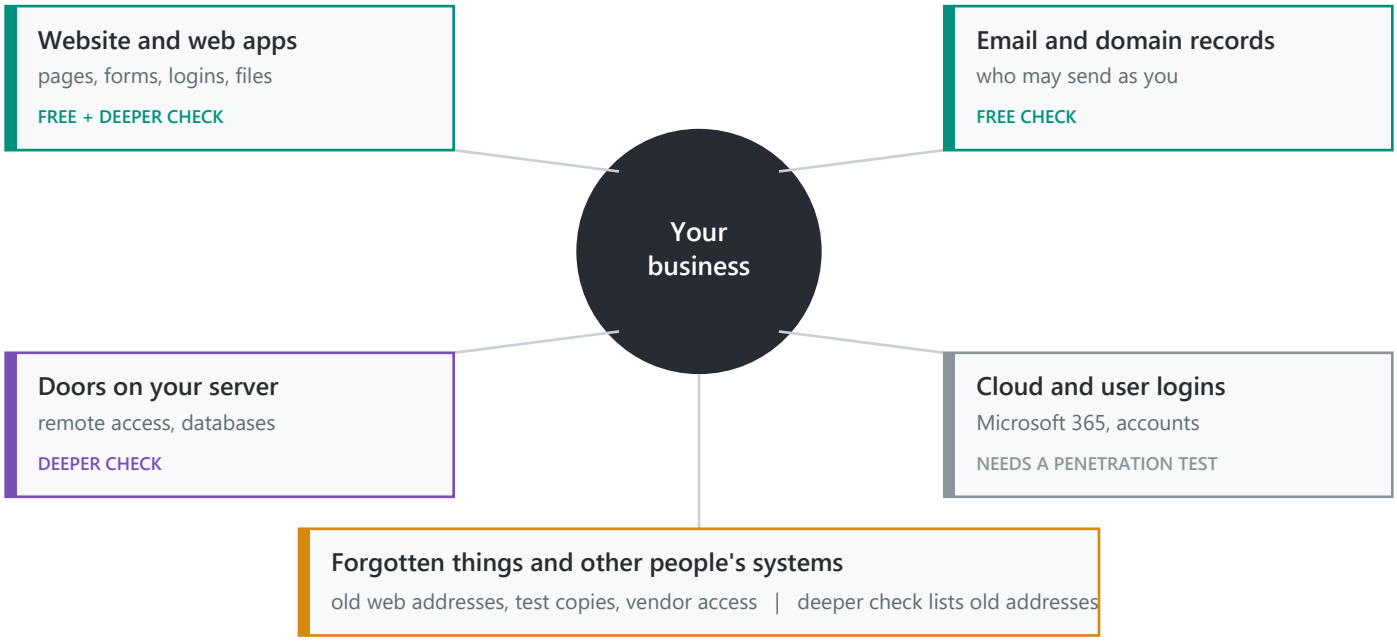


Figure 2. The 5 parts of a small business's outside, and what checks each one.

Why look at the outside first

- **Nothing to install, nothing to hand over.** The free check needs your domain name. That's all. No software, no password, no change to your systems.
- **No disruption.** It sends normal requests, slowly. It stops when it sees a sign of a problem. Your staff won't notice it.
- **About you, by name.** The result names your domain and your records. It turns "we're probably fine" into a list you can act on.
- **One list for everyone.** Often the site, the domain, the host and the email belong to different people. One report puts them all on the same page.

> **Things we often find on the outside**
>
> A test copy of the site that still works. An old web address that points at a service you stopped paying for. A backup file left on the site after a move. A door opened for a vendor years ago. An add-on nobody has updated since the developer left. Each is easy to fix once someone knows it is there.

SECTION 5

Reading your result

Your result has four parts. A grade. A score out of 100. A list of findings, each with a level of seriousness. And a list of checks that could not run.

> **In plain words**
>
> Each problem costs points. The worst problem also limits your best grade. "Could not check" never counts as a pass.

How serious is each finding?

Every finding has one of five levels. Each problem takes points off a starting score of 100. The worst problem also sets the best grade you can get. That way one big problem can't hide behind many small passes.

SEVERITY	POINTS OFF, EACH	BEST GRADE	EXAMPLE FINDING
CRITICAL	-40	Grade F	Environment file is public
HIGH	-20	Best grade C	No DMARC policy
MEDIUM	-8	Best grade B	HSTS is not set
LOW	-3	No limit	No CAA record
INFO	0	No points off	DNSSEC isn't enabled

Figure 3. Each problem costs points. The worst one sets your best possible grade.

The grade

Your score puts you in a grade band. Then the limit from your worst problem applies. You get whichever grade is lower.



The grade comes from **21 main checks** in the free check. If fewer than 17 of them could run, you get no grade. The result tells you how many ran. Problems found by the deeper check count toward the score too.

A worked example

Here is a typical free-check result for a small business. The titles are the ones the scanner uses.

Finding	Level	Points
DMARC only monitors, it doesn't block	Medium	-8
No Content-Security-Policy header	Medium	-8
No Permissions-Policy header	Low	-3
Server announces its software	Low	-3
No CAA record	Low	-3
DNSSEC isn't enabled; MTA-STS isn't set up; No security contact is published	Info	0
Score		**75**

A score of 75 is in the C band. The worst problem is medium. Medium limits the grade to B. C is lower, so the grade is **C, 75 out of 100**. Now fix the two medium problems. The score goes up to 91. The worst problem is now low. The grade becomes A.

Two more examples. A site whose only problem is **No DMARC policy** scores 80. That is the B band. But a high problem limits the grade to C. So the grade is C. A site whose only problem is **Environment file is public** scores 60. A critical problem means F.

> **"Could not check" is never a pass**
>
> Sometimes a site blocks the scan. Or a firewall drops the connection. Or a lookup takes too long. The scanner shows that as its own result, with the reason. It never counts it as a pass. It also lowers how many of the 21 checks ran. If you see one, find out why. Often the news is good. Your firewall did its job. But you should know, not guess.

What is at stake

Computer programs test public websites all day. Your site doesn't need to be famous to be tested. It only needs to be online. When a site has a gap, these are the usual results:

- **Data is taken.** A forgotten backup file can hand over customer records. So can an open database door. Nobody even has to guess a password.
- **Someone sends email as you.** Without strong email records, a fake bill can carry your exact domain name. Your customers are the ones who get it.
- **Your site is used against your visitors.** It is changed, or made to send visitors somewhere harmful. Or it helps a fake

link look like yours.

- **A way into the rest of the business.** Open login pages, old software and open remote access are used to get inside. Ransomware gangs often start this way. They lock your files and ask for money.
- **A question you can't answer.** A customer, an insurer or an auditor asks about your security. You have nothing to show them.

A gap does not mean any of this will happen. It means one less thing is in the way. Most gaps are cheap to close once you know about them. Finding out afterwards costs more.

Findings as the PDF report shows them

Each finding is a card. The coloured label shows how serious it is. Next to it are the area and which check found it. Then comes the title, what we saw and how to fix it. Findings from the deeper check add one more line. It shows the exact request we sent and what came back. Your developer can repeat it.

HIGH Open ports | verified check

Windows Remote Desktop (port 3389) answers from the internet

Open Remote Desktop is one of the most common ways ransomware gangs get in. We only opened a connection and closed it again.

How to fix: If nothing outside your network needs it, close the port at the firewall, or allow only your own addresses or a VPN.

EVIDENCE

```
TCP connect to 203.0.113.10 port 3389 succeeded; nothing was sent
```

INFO Admin and login pages | verified check

The WordPress login page is public

Anyone on the internet can open /wp-login.php on this site. Attackers look for these pages first and try leaked or common passwords on them.

How to fix: If this page doesn't need to be public, allow it only from your office or VPN, or put it behind single sign-on. Keep the software up to date and use strong, unique passwords with multi-factor sign-in.

EVIDENCE

```
GET /wp-login.php -> 200, 4211 bytes, title: Log In - Example Shop
```

Figure 4. Two sample findings, drawn like the PDF report. 203.0.113.10 is a made-up address.

The report comes in this order. First, **What needs attention**, worst first. Then **Worth knowing**. These are small items that don't cost points. Then **What we couldn't check**. Then **Everything we checked**, with every result. It ends with what the check does and what to do next. The results page uses the same levels and titles.

Website problems are often nobody's job

Marketing owns the site. A developer built it. A hosting company runs it. Someone else holds the domain. The IT provider looks after email. The scan crosses all of those lines. Each card in section 6 tells you who to call.

SECTION 6

The 16 kinds of problem, in plain words

These are the 16 kinds of problem the scan can find. Each card says what it is and how an attacker uses it. It shows what your report may say. It gives the usual fix, who does it and how long it typically takes.

In plain words

Find your report's finding in the cards below. Then send the card to the person named under "Who fixes it".

Free-check cards have a grey bar. Deeper-check cards have a purple bar. The times are typical guesses for planning. They are not quotes or promises. Your set-up and your provider's workload will change them.

1. The lock on your site (encryption and certificates) FREE CHECK

In plain words: the padlock in the browser must be current, real and strong.

What it is. The padlock means the connection to your site is scrambled, so others can't read it. The technical name is TLS. The certificate is the document that proves the site is really yours. The scan reads your certificate. It checks that your server refuses old versions of the lock (TLS 1.0 and 1.1). It checks that it refuses broken ways of scrambling (RC4 and 3DES). It also checks that the plain http address sends visitors to the locked https address.

How an attacker uses it. On shared Wi-Fi, a weak lock makes it easier to read or change what visitors send. An expired certificate trains visitors to click past warnings. That is exactly what a fake site needs.

Your report may say. **Certificate has expired** (critical). **Certificate expires very soon** (high, under 14 days). **Certificate expires within a month** (low). **Certificate doesn't match the domain** (high). **Certificate isn't trusted** (high). **Server only works with old TLS versions** (high). **Server accepts old TLS versions** (medium). **Server accepts broken ciphers** (medium). **http does not redirect to https** (medium).

The usual fix. Renew the certificate. Turn on automatic renewal. Allow only the current versions of the lock. Send every http visit to https. With many hosts these are switches, not code.

Who fixes it: your web host or developer. **Typical time:** 15 to 60 minutes for a certificate. Up to half a day to turn off old versions and test.

2. Settings your site sends to browsers (security headers) FREE CHECK

In plain words: your site should tell browsers to use the safety features they already have.

What it is. With every page, your site sends short instructions to the browser. These are called headers. Some turn on safety features. One says "always use the locked address" (HSTS). One says "only run code from places I trust" (Content-Security-Policy). Others stop other sites showing your pages inside theirs. They stop browsers guessing file types. They keep full page addresses private. They switch off things like the camera that pages don't need. The scan also checks which other websites may read your pages (CORS).

How an attacker uses it. Missing settings make some tricks easier. One trick hides your real page under a fake one, so a visitor clicks the wrong thing. Another runs a harmful script as if it were yours.

Your report may say. **HSTS is not set** (medium). **HSTS lifetime is short** (low, under 6 months). **No Content-Security-Policy header** (medium). **Other sites can put yours in a frame** (low). **Browsers may guess file types** (low). **No Referrer-Policy header** (low). **No Permissions-Policy header** (low). **Any website can read this response** or **Wildcard CORS is combined with credentials** (low).

The usual fix. Add a standard set of these settings. Your host's control panel often has them. Turn on Content-Security-Policy in "report only" mode first. See what it would block. Then switch it on fully.

Who fixes it: your developer or web host. **Typical time:** 1 to 2 hours for the simple settings. A few days to a few weeks for Content-Security-Policy on a busy site, because it needs testing.

3. Cookies FREE CHECK

In plain words: login cookies need three safety flags, like a lock, a seal and a "do not share" sticker.

What it is. A cookie is a small note your site leaves in the visitor's browser. A login is often kept in a cookie. Three flags protect cookies. Secure means only send it over the locked connection. HttpOnly means scripts can't read it. SameSite means don't send it when another site asks. The scan reads the cookies your home page sets.

How an attacker uses it. If a script can read a login cookie, or it travels unlocked, it can be stolen. A stolen login cookie can work like a stolen password.

Your report may say. **Cookies can travel over unencrypted connections**, **Scripts can read these cookies** (low, or medium when the name suggests a login). **Cookies are sent with requests from other sites** (low). The finding names the cookies.

The usual fix. Set Secure on every cookie. Set HttpOnly on every cookie scripts don't need. Set SameSite to Lax, or Strict where it fits.

Who fixes it: your developer, or the settings of your site's add-ons. **Typical time:** 1 to 4 hours, including a test of logins and checkout.

4. Protection against fake email from your domain (SPF and DMARC) FREE CHECK

In plain words: two records tell the world which email really comes from you, and what to do with fakes.

What it is. Two records protect your name in email. The first lists the servers allowed to send email as you. It is called SPF. The second tells other mail servers what to do with email that fails the check. It is called DMARC. It can say "just report it", "put it in spam" or "refuse it". The scan reads both. It drops a leading www, so www.example.com is checked as example.com.

How an attacker uses it. Without a strong DMARC record, anyone can send email that shows your exact domain. That is how fake bills and fake "change our bank details" emails reach your customers and staff.

Your report may say. **No DMARC policy** (high). **SPF lets any server send as your domain** (high). **No SPF record**, **More than one SPF record**, **SPF doesn't say whether other servers are allowed**, **DMARC only monitors, it doesn't block**, **More than one DMARC record**, **DMARC record has no policy**, **DMARC policy isn't valid** (all medium). **SPF record has no closing rule**, **DMARC applies to only part of your mail** (low).

The usual fix. Publish one SPF record. It must list every service that sends email for you. Start DMARC in "report only" mode with an address for the reports. Read the reports for a few weeks. Then move to "spam", and finally to "refuse".

Who fixes it: your IT provider or whoever manages your domain. **Typical time:** about 1 hour to publish the records. Then 2 to 6 weeks to reach full protection, because you must first find every service that sends as you.

5. Locked delivery of your incoming email (MTA-STS) FREE CHECK

In plain words: this asks other mail servers to always use a lock when they send email to you.

What it is. A record plus a small rules file at `mta-sts.yourdomain`. It tells other mail servers to lock email on its way to you. If they can't, they should not deliver it.

How an attacker uses it. Without it, someone between two mail servers can remove the lock and read the email. This is harder than most attacks here. That's why a missing rules file is "info" only.

Your report may say. **MTA-STS isn't set up**, **MTA-STS is switched off** (info). **MTA-STS policy file can't be read**, **MTA-STS is in testing mode**, **MTA-STS policy has no mode** (low).

The usual fix. Publish the record and the rules file in testing mode. Read the reports. Then switch the mode to enforce.

Who fixes it: your IT provider. **Typical time:** 2 to 4 hours to set up. Then 2 to 4 weeks in testing mode.

6. Who may make certificates for you (CAA) FREE CHECK

In plain words: a record that names the only companies allowed to make a certificate for your domain.

What it is. Certificates are made by a short list of trusted companies. A CAA record names the ones you use.

How an attacker uses it. Without it, any of those companies may make a certificate for your name. The record makes it harder to get one for a fake copy of your site.

Your report may say. **No CAA record** (low).

The usual fix. Ask your host which company makes your certificates. Then publish a record that names it, for example `0 issue "letsencrypt.org"`.

Who fixes it: whoever manages your domain. **Typical time:** 15 to 30 minutes.

7. A seal on your domain records (DNSSEC) FREE CHECK

In plain words: a seal that shows your domain's records really came from you.

What it is. A digital seal on your domain's answers. It lets others tell a real answer from a fake one.

How an attacker uses it. Without the seal, a fake answer could send visitors or email to the wrong place. It is harder to spot. This attack is rare against small sites, so the finding is "info" only.

Your report may say. **DNSSEC isn't enabled** (info).

The usual fix. Turn on DNSSEC with whoever hosts your domain records. Then add the code they give you at the company where you bought the domain. Some providers do both in one click.

Who fixes it: whoever manages your domain. **Typical time:** 1 to 2 hours of work. Then up to a day or two to take effect.

8. A "report a problem" contact (security.txt) FREE CHECK

In plain words: a small file that says who to tell if someone finds a problem on your site.

What it is. A small text file at `/.well-known/security.txt`. It tells helpful researchers how to reach you.

How an attacker uses it. They don't. The risk is the other way round. A helpful person finds a problem but has nobody to tell. So the problem stays open.

Your report may say. **No security contact is published** (info).

The usual fix. Publish the file with a Contact line, such as an email you check.

Who fixes it: your developer or web host. **Typical time:** about 30 minutes.

9. Private files left in public FREE CHECK

In plain words: like leaving a spare key and the alarm code under the doormat.

What it is. Some files should never be public. One is the hidden folder a developer's tools use (`.git`). One is a settings file full of passwords (`.env`). Others are database copies, backups, a page of server settings and folders that list their own contents. The scan asks for 10 of the most common names. It checks that what comes back really is that kind of file. It also notes when your server tells the world its software and version.

How an attacker uses it. These files can hand over passwords, code or customer data with no effort. A version number tells an attacker which known weaknesses to try.

Your report may say. **Source control folder is public**, **Environment file is public** (critical). **Database dump is public**, **Backup archive is public**, **PHP configuration page is public** (high). **Server status page is public**, **Directory listing is on for /uploads/** or **/backup/** (medium). **Server announces its software** (low). **Technologies we could identify** is for your information only. It never costs points.

The usual fix. Remove the file and block the address. Then assume someone has read it. Change every password and key that was inside. Turn off folder lists and version numbers.

Who fixes it: your developer or web host. **Typical time:** under 1 hour to remove. Up to a day to change the passwords inside.

10. Login and admin pages anyone can open DEEPER CHECK

In plain words: the staff door to your site is on the street. Anyone can try a key.

What it is. These are the pages where you or your staff log in to run the site. Some tools should never be on the internet at all, such as a database tool. The scan asks for 26 common addresses, such as `/wp-login.php` and `/phpmyadmin/`. It counts a page only if it is a real page of its own. It reads the page and stops. It never tries to log in.

How an attacker uses it. Public login pages get password guesses all day. The guesses often use passwords leaked from other sites. A database tool on the internet is one weak password away from your data.

Your report may say. The page's name followed by "is public". For example, **The WordPress login page is public** (info). Login pages are often meant to be open, but each one is a target. Or **phpMyAdmin, a database admin tool is public** (low). The last line shows the address, the answer and the page title.

The usual fix. If a page doesn't need to be public, allow it only from your office or a private connection. Remove tools you

don't use. Use strong, different passwords. Turn on a second step at login, such as a code on your phone.

Who fixes it: your developer or IT provider. **Typical time:** 1 to 4 hours. The second login step for a few users is often done the same day.

11. Old software with known weaknesses DEEPER CHECK

In plain words: old software is like an old lock that burglars already know how to open.

What it is. Your site often tells visitors which software and version it runs. That includes the web server, PHP, and site builders like WordPress, Drupal or Joomla. It also includes 3 common code libraries: jQuery, Bootstrap and AngularJS. The scan checks these against a short list of known weaknesses. IA checks that list by hand. The scan sends no extra requests for this. It reads the home page it already loaded.

How an attacker uses it. Attackers search for sites by version number. Then they use weaknesses that are already public. This is one of the most common ways small sites are taken over.

Your report may say. The product and version, followed by **has known issues** (up to high) or **no longer gets security fixes** (medium). The finding lists the public ID numbers of the known weaknesses. A match means the version has known weaknesses. It does not prove they work on your site. Your host may already have patched them.

The usual fix. Take a backup. Update to a supported version. Remove add-ons and code you don't use. Turn on automatic updates where it is safe.

Who fixes it: your developer or web host. **Typical time:** 1 to 4 hours for small updates. 1 to 5 days for a big upgrade that needs testing.

12. Pages that trust what visitors type DEEPER CHECK

In plain words: a page should treat anything a visitor types as plain text, never as instructions.

What it is. Some pages show visitor text back on the page as code. Some put it straight into the database. Both are weaknesses. The scan finds up to 10 inputs in links and search boxes on your home page. It sends each one a harmless test word with the symbols < > " '. It sends each one more request with a single quote mark added. It only reads. It never sends a form that uses POST.

How an attacker uses it. In the first case, someone can send your visitors a link that runs their code on your site. In the second, someone may read or change the data behind your site.

Your report may say. **A page shows visitor text without escaping it** (medium). The test word came back exactly as sent. **A page shows a database error when given a quote** (high). A known database error appeared that the normal page doesn't show.

The usual fix. Fix it in the code. Make every page treat visitor text as plain text. Make every database call keep visitor text separate. Turn off detailed error messages on the live site. A web firewall can help while the fix is made. Then test again.

Who fixes it: your developer. **Typical time:** 1 to 5 days, including a retest. Treat a database error as urgent.

13. Old, forgotten web addresses DEEPER CHECK

In plain words: an old sign that still points to an empty shop. Someone else could move in.

What it is. You may have addresses under your main one, such as shop.example.com. These are called subdomains. The scan looks up 40 common names. It only looks them up. It never connects to them. It lists the ones that exist. Then it checks whether any of them still point to an outside service that no longer answers.

How an attacker uses it. Say an old address still points to a service you stopped using. Anyone who signs up there with the same name can put their own page on your address. That makes a very convincing fake page.

Your report may say. **Subdomains we could find** (for your information). **A subdomain points at a service nobody has claimed** (high).

The usual fix. Delete the old record. Or claim the name back at the service. Keep a list of every address and who owns it.

Who fixes it: whoever manages your domain. **Typical time:** 15 minutes to 1 hour.

14. Open doors on your server (ports) DEEPER CHECK

In plain words: your server has numbered doors. Only the ones you use should be open.

What it is. A server has numbered doors called ports. Each service listens behind one. The scan knocks on 25 common doors. It tries up to 2 of your proven addresses. It opens the door and closes it at once. It sends nothing and reads nothing. First it knocks on one random door that nobody should use. If that opens too, something answers on every door. Then the scan says it couldn't tell.

How an attacker uses it. Database, remote desktop, file sharing and remote control doors are tried first. Open remote desktop is one of the most common ways ransomware gangs get in.

Your report may say. The service and door, then "answers from the internet". For example, **Windows Remote Desktop (port 3389) answers from the internet**. Databases, file sharing and remote control are high. File transfer (FTP) is medium. SSH, email and the extra web doors 8080 and 8443 are info.

The usual fix. Close the door at the firewall. Or open it only for your own addresses or a private connection. If it must stay open, keep it updated and use strong logins.

Who fixes it: your IT provider or hosting company. **Typical time:** 30 minutes to 2 hours for a firewall rule. 1 to 5 days to move remote access behind a private connection.

15. Links that bounce visitors to any site DEEPER CHECK

In plain words: your site should not forward visitors to any address a stranger chooses.

What it is. Some sites take an address from a link, such as `?next=`, and send the visitor there. The scan tries 6 common names. It uses an address that can never exist (`canary.invalid`). It never follows the link.

How an attacker uses it. A fake email links to your real domain. Your site then forwards the reader to the attacker's site. The link looks like yours because it starts with yours.

Your report may say. **The home page redirects to any site you name** (medium).

The usual fix. Only forward visitors to your own pages, or to a fixed list of sites.

Who fixes it: your developer. **Typical time:** 2 to 8 hours.

16. Risky request types (HTTP methods) DEEPER CHECK

In plain words: your server should not accept requests that change or delete files, or echo back what it receives.

What it is. Every web request has a type. GET reads a page. PUT and DELETE change or remove files. TRACE echoes the request back. The scan sends 1 OPTIONS request to ask which types your home page accepts. It sends 1 TRACE request. It never sends PUT or DELETE.

How an attacker uses it. If PUT or DELETE really work without a login, anyone could change your files. TRACE, together with another weakness, can leak login details.

Your report may say. **The server advertises methods that change content** (low). **TRACE is switched on** (medium).

The usual fix. Switch off PUT, DELETE and TRACE, unless something on your site needs them.

Who fixes it: your web host or developer. **Typical time:** 30 minutes to 1 hour.

SECTION 7

What a scan can't see, and what a penetration test finds

A scan points at doors that may be unlocked. A penetration test tries the handle, with your written permission. Then it writes down what was behind the door. Both are useful. They answer different questions.

> **In plain words**
>
> The scan looks from the street. A test, with your OK, checks whether someone could really get in.

Where the scan stops

Area	Why the scan can't judge it
Logins	The scan never logs in. It can't test password rules, lock-outs or the second login step.
Pages behind a login	Customer portals, booking systems and admin screens sit behind a login. The scan only sees the front door.
How your site is meant to work	Can one customer see another's records? Can a discount be used twice? The scan doesn't know your rules.
Small problems that add up	Two small findings can join into a real way in. The scan rates each finding on its own.
The inside and user accounts	Your office network, user accounts, Microsoft 365 settings and cloud storage are out of its reach.
Human judgment	Does an open page hold something private? Does a finding matter for you? That takes a person.

Four made-up examples

These examples are invented. They show the kind of thing a test finds and a scan can't. They are not real cases. They do not describe any real business.

1. The invoice number

A firm's client portal shows each bill at an address that ends in a number. A tester logs in as a test client. They change the number by one. Now they see a different client's bill, with name, address and amount. From the outside, the portal looked fine. The flaw was in what the portal allowed once you were inside.

2. Two small findings that add up

A scan finds a public WordPress login page. It is marked "info". Nothing else stands out. During an approved test, the tester finds two more things. Staff logins have no second step. And the login allows endless guesses. Together, that is a real way into the admin area. Each piece alone looked small.

3. The test copy of the site

The deeper check lists staging.example.com among the old addresses it found. The scan stops there. A tester looks closer. They find an old copy of the site. Its password was shared with everyone at the agency. It runs last year's add-ons. And it is linked to the same customer database as the live site.

4. The shared folder

A website scan comes back clean. Then a test of the company's Microsoft 365 finds two problems. A folder of payroll files is shared with "anyone with the link". And a former worker's account can still log in. Neither shows on the website. Both are found by someone who checks user and cloud settings, with permission.

The short version

The scan tells you what might be wrong from the outside. A penetration test shows what someone could really do. It gives you proof for each finding. Section 11 explains when you need one, and how IA does them with Argos Phantom.

SECTION 8

Your fix plan

Sort by what an attacker could do, not by how long the list is. Close the open doors first. Then fix the weak locks. Then do the tidying up.

In plain words

Fix the worst things this week. Fix the middle things this month. Tidy up the rest this quarter. Check again after each fix.

This week

OPEN DOORS

- Critical and high findings.
- Private files: remove them. Change the passwords.
- Close open database and remote desktop doors.
- Start email protection (DMARC).

→

This month

WEAK LOCKS

- Medium findings.
- Update software. Back up and test first.
- Hide login pages or add a second login step.
- Make every visit use the lock.

→

This quarter

TIDYING UP

- Low and info findings.
- Turn email protection up to full.
- Add the small domain records.
- Write down who owns what.

Each time frame in more detail

This week is for anything critical or high. It is also for any private file left in public, even one marked medium. Add any public database tool or open remote desktop door. Add a missing DMARC record. Publish one in "report only" mode today, so the reports start coming. And renew any certificate that runs out within a month.

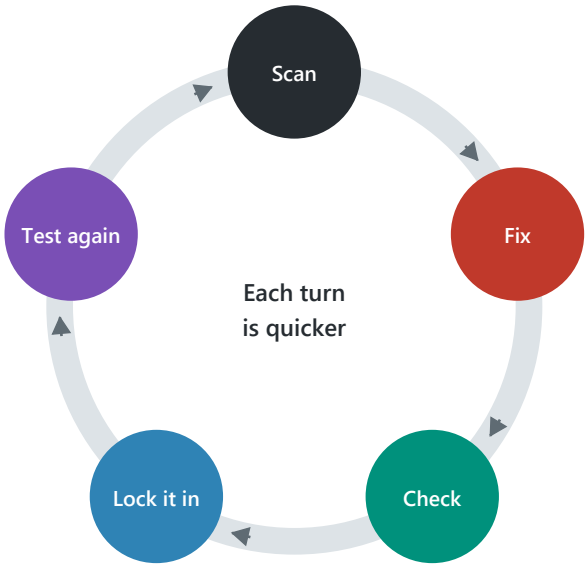
This month is for the medium items and software updates. Updates need care. Take a backup first. Update. Test the pages that matter, like forms, checkout and logins. Then scan again.

This quarter is for the low and info items. Move your email protection from "report only" to "spam", then "refuse", as the reports allow. Write down who owns the site, the hosting, the domain and email. This is also the time to decide whether you need a penetration test (section 11).

Lock it in means writing down the settings you fixed. That means your browser settings, lock versions, email records and closed doors. They become your own standard. Then the next site update or new vendor can't quietly undo them. **Test again** means a deeper check each quarter. When the signs in section 11 apply, it also means a penetration test.

Fix, check, repeat

Fixing once is not the goal. Staying fixed is. Do the same five steps every time. Each round is quicker than the last.



1 Scan
See what the outside shows

2 Fix
Worst first. Owner and date.

3 Check
Prove each fix

4 Lock it in
Write the settings down

5 Test again
Deeper check, pen test

Is a finding wrong for your set-up? Don't just ignore it. Write down why, who decided and when. Use the worksheet at the back of this guide. Look at it again next quarter. The checklist after the worksheet lists every step.

SECTION 9

Emails you can copy

Here are 4 emails you can copy, change and send. Replace everything in **[brackets]**.

> **In plain words**
>
> Use these so you don't have to write the hard emails yourself.

A. Sending the report to your developer or web host

> EMAIL TEMPLATE
>
> **Subject:** Website security scan of [domain]: findings to fix
>
> Hi [name],
>
> We ran an outside security scan of [domain] on [date]. The results are attached. The scan only reads. It came from IA's Website Security Check. In your logs it shows as IA-SiteCheck/1.0.
>
> Could you please:
>
> 1. Confirm each finding marked critical or high. Tell me if you can't see one.
> 2. Give me a date to fix those. Give me a second date for the medium items.
> 3. Tell me before you change anything that could break the site, so we can test first.
> 4. Tell me which items you think are wrong for our set-up, and why.
> 5. Tell me when you're done. I'll scan again and send you the new result.
>
> If any of this is outside what we pay you for, tell me the cost and the time.
>
> Thanks,
> [your name]

B. Asking permission to scan a site that isn't yours

Get written permission before you scan a site you don't own or manage. This applies even to the free check. Scanning without it can break contracts. In some places it can break the law.

> EMAIL TEMPLATE
>
> **Subject:** Permission to run a security scan on [domain]
>
> Hi [name],
>
> May [your name or company] run an outside security scan of [domain] on or after [date]? It looks for security problems anyone on the internet can see.
>
> The first check sends about 15 normal requests. It needs no login. It doesn't try passwords, send forms or change anything. I'll send you the results.
>
> A deeper check is possible later. But someone at [domain] must first prove they control the domain. They do this with a small file, a page tag or a domain record, using a work email at [domain].
>
> Please reply "Approved" to go ahead.
>
> [your name]
> [your company, phone]

C. Answering an insurer or customer who asked about your security

Say what you checked and what you're doing. Don't claim more than the scan shows.

> EMAIL TEMPLATE
>
> **Subject:** Re: [their question or form name]
>
> Hi [name],
>
> Thank you for asking. On [date] we ran an outside scan of our website, [domain]. It checked the lock on our site, browser settings, cookies, email protection, domain records and private files. [Add if true: We also ran a deeper check of login pages, software versions, old web addresses and common doors.]
>
> It found [number] items to fix. [Number] were high or critical. We fixed [what] on [date]. A second scan confirmed it. The rest are planned for [month]. [Owner] is in charge of them.
>
> This was an outside scan. It was not a penetration test or an audit. [Add other steps you really take, such as a second login step for email.] I'm happy to share the report or answer more questions.
>
> Kind regards,
> [your name, title]

D. A short update for your boss or board

> EMAIL TEMPLATE
>
> **Subject:** Website security: where we are
>
> [Name],
>
> Where we started: grade [grade], score [score] out of 100 on [date]. We had [number] high or critical items.
>
> Where we are now: grade [grade], score [score] on [date]. Fixed: [one line each]. Still open: [one line each, with owner and date].
>
> What I need: [a decision, a budget line, an introduction, or nothing].
>
> Next scan: [date]. The scan checks what anyone can see from the internet. It doesn't prove we are safe. It doesn't cover email accounts, computers or the office network.
>
> [your name]

SECTION 10

If your result is clean

A clean result is good news. Most businesses can't show one. It is also a snapshot of one day. Keep it, and keep it up to date.

> In plain words
>
> Clean today doesn't mean clean next month. Keep the report and scan again after any change.

What clean means

Every check that ran found nothing to fix. Enough checks ran to give a grade. From the outside, on that day, nothing obvious was open or set up wrong, in what we checked.

Three things a clean result gives you

- **Proof instead of a guess.** Most owners think their site is "probably fine". A clean, dated report shows it, for what was checked. That helps when a customer or insurer asks.
- **A starting point you can trust.** We compare the next scan with this one. If an update or a new vendor opens a gap, you see it right away. You don't find out months later.
- **A reason to keep checking.** Most problems start with a change nobody checked. A clean result is the best time to set a regular scan, while nothing is urgent.

Someone may ask what it means. You can say: "We scanned on [date]. From the outside, it found nothing to fix. We keep the report. We scan again after changes." It is honest. It doesn't claim more than the scan shows.

What clean doesn't mean

- It doesn't cover what happens after someone logs in. It doesn't cover flaws in how your site works, like a checkout that can be tricked. That needs a person.
- It doesn't cover your email accounts, Microsoft 365, staff passwords, computers or office network. Attackers often go there next. Our free Microsoft 365 checklist covers part of that: intelamation.com/m365-security-checklist.html.
- It covers one domain. Your other domains, and sites your vendors run for you, need their own scans.
- "Could not check" items are not clean. Find out why they couldn't run.

How to keep it clean

- Keep the report as your starting point. Compare the next one with it.
- Keep a short list of changes. Note new add-ons, new forms, new web addresses, and a new host or email provider.
- Keep automatic certificate renewal on. Put a real person's name on the renewal emails.
- Remove add-ons and test copies of the site you don't use.
- When you stop using a vendor or service, delete its domain records that same day.

When to scan again

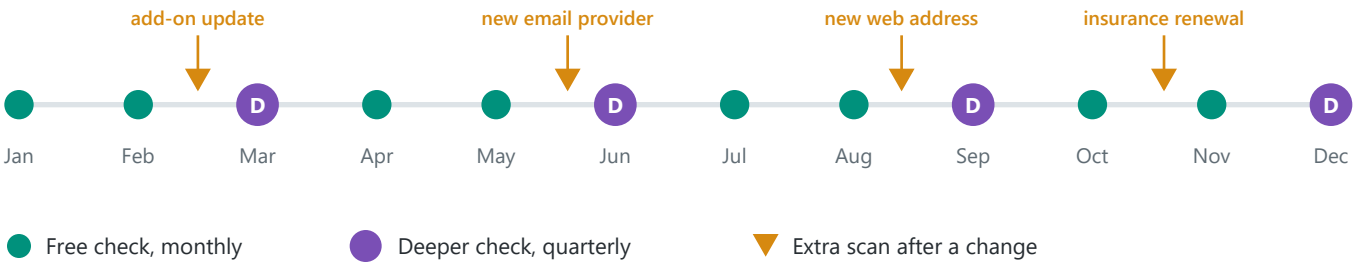


Figure 5. When to scan again, for most small sites. The events are examples.

For most small sites, scan once a month. Scan again after any big change. Run the deeper check every quarter. Also run it before an insurance renewal or a customer's security form, so your answer is recent.

SECTION 11

When you need a real penetration test

A penetration test is a planned attempt to break in. Qualified people run it or oversee it. They stay within limits you agree in writing. It finds what scans can't. It also costs more and takes longer. So it helps to know when you need one.

> **In plain words**
>
> Do you take payments or keep private records behind a login? Has someone asked you for a test? Then you probably need one.

Signs you need one

- Your site takes payments or has customer accounts. Or it handles health, legal or money records.
- A contract, a regulator, an insurer or a large customer asks for one.
- You've built your own features, like a customer portal, a booking system or an app.
- You've had a security problem. Or a scan keeps finding pages that trust what visitors type.
- You're about to launch something new and important.

What kinds there are

- **Outside.** Everything your business shows the internet: sites, remote access, email and firewalls.
- **One website or app, in depth.** Usually logged in as each kind of user. It checks logins, who can see what, forms, payments and how the app is meant to work.
- **Inside.** What someone could reach after getting into your network, for example through one infected computer.
- **Cloud and Microsoft 365.** The settings of your cloud accounts. Who can log in, from where, how, and what they can reach.

How to read a quote

- **What is included.** The quote should list exact addresses, apps and kinds of user. "Your website" is not enough.
- **How findings are proven.** A list of possible problems is a scan. A test shows what could really be done, with proof for each finding.
- **How often.** A one-time test shows one day. Ask whether testing repeats, and how often.
- **The report.** Look for a short summary for owners. Look for findings with steps to repeat them. Ask for a sample.
- **Testing again.** Is a second test of fixed items included? Within what time?
- **Rules.** When testing happens, what is off limits and how urgent problems are handled.

8 questions to ask before you hire a tester

1

What exactly will you test?

A GOOD ANSWER
A written list of addresses, apps and kinds of user.

2

How do you prove a finding?

A GOOD ANSWER
Each one comes with proof. Not just a tool's guess.

3

Who runs and oversees it?

A GOOD ANSWER
Named, qualified people who answer for the test.

4

Will you test logged in?

A GOOD ANSWER
Yes. With a test account for each kind of user.

5

How do you avoid breaking the site?

A GOOD ANSWER
Written rules, agreed times and a person who can stop it.

6

Can we see a sample report?

A GOOD ANSWER
A short summary, plus findings with steps to repeat them.

7

Do you test again after we fix?

A GOOD ANSWER
Yes. Within an agreed time.

8

How do you protect our data?

A GOOD ANSWER
Written data rules, and insurance for their work.

Figure 6. Ask every tester the same 8 questions. Then compare the answers.

How to get ready

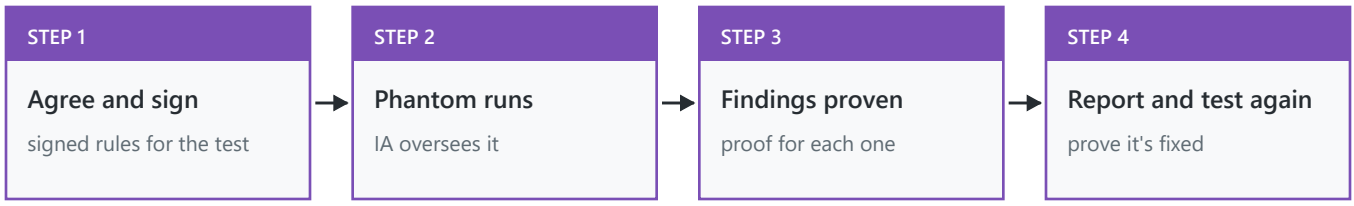
- Get written permission from someone who can give it. Check whether your hosting or cloud provider needs to know too.
- Run this scan first and fix the easy findings. Then the test can focus on the hard ones.
- Make test accounts for each kind of user. Never hand over a real customer's login.
- Take backups. Tell your IT provider when testing will happen.
- Agree a contact on both sides for the testing days. Pause big changes to the site.

About Argos Phantom

Argos Phantom is IA's penetration testing service for small and mid-sized businesses. It runs automatic tests on a regular schedule. IA's security team oversees every test. You find out what an attacker could do before one tries.

- **Approved and limited.** Every test starts with signed rules and a written list of what may be tested, by whom and when.
- **Safe by design.** It only reads and checks settings. It is built never to delete data, lock accounts or change records. Every action is written to a log.
- **Proof, not guesses.** A finding counts only once it is confirmed. Each one comes with proof, what it could reach and how to fix it.
- **Regular.** It looks for ways small problems join up into a way in. It runs again on a schedule. So fixes get checked, and new gaps show up.

You choose what Phantom tests. It can test what faces the internet. It can test your office network and the system that runs your Windows logins (Active Directory). It can test cloud accounts in AWS, Azure, Google Cloud and Microsoft 365. It can test user login settings in Microsoft 365 and Google Workspace. With your OK, it can run fake phishing emails. These only record whether someone clicked. They never collect a real password. You get one report, sorted by what matters most. The results can also feed your compliance records in Argos GRC.



Then it runs again on a schedule. New gaps show up. Fixes stay proven.

What we ask when you request a test

- What to test: the outside, the office network, cloud and Microsoft 365, user logins, email and phishing, or "not sure".
- About how many internet addresses or servers you have, and how many web apps.
- Whether a compliance or insurance rule is behind the request.
- When you need it.

Phantom plans start at $250 a month. We agree the scope and the final price with you in writing before any testing starts. Request a test at phantom.intelamation.net, or call (888) 711-4521.

SECTION 12

Your next steps with IA

You can fix everything in your report yourself. Your own developer and this guide are enough. If you'd like help, we work with businesses in three ways. The first one is free.

In plain words

Talk to us for free. Then decide if you want us to fix things or test more deeply.

1

Free findings review

45 MINUTES

We go through your result. We sort it by week, month and quarter. We tell you who should fix what.

2

Fixed-scope fix project

WRITTEN SCOPE, FIXED PRICE

We fix the agreed findings with your host and developer. We scan again. You get a before-and-after.

3

Argos Phantom penetration test

APPROVED IN WRITING

A test that shows what an attacker could really do. You get proof for each finding and a second test.

What happens in a free findings review

It is a free 45-minute video or phone call with an IA security lead. Bring your report. Or tell us your domain, and we'll look at the free-check result together. Do you run a law firm, a medical practice or an accounting firm? Then ask for the free review named on our page for your field. That is the technology-competence review, the HIPAA posture review or the Safeguards Rule readiness review. It also takes 45 minutes, and it covers your scan result too.

What we ask you	What you leave with
Who looks after the site, the domain and email today?	Your top 3 findings in plain words. What each means and who usually fixes it.
What changed lately? A new site, host, add-on or vendor?	A short plan for this week, this month and this quarter.
Does the site take payments, logins or private forms?	The email to send your developer or host, ready to go.
Has a customer, insurer or auditor asked about security?	An honest view on whether you need a penetration test, and what kind.

There is no obligation. Sometimes the right answer is "your developer can fix this in an afternoon". If so, that's what we'll tell you.

A fixed-scope fix project

Maybe you'd rather hand the work over. We agree a written list of findings to fix. We do the work with your host and developer. We scan again at the end. We give you a before-and-after summary. The scope and price are fixed before we start. Afterwards, we can keep checking every month, so the result stays current.

A penetration test with Argos Phantom

Do the signs in section 11 apply? Or has a customer, insurer or auditor asked for a penetration test? Then we plan and approve a Phantom test with you. You get a written report with proof for every confirmed finding. We walk you through it. Then we test again to show your fixes worked.

Find out what an attacker would see, before one looks.

Book a free 45-minute findings review. Bring your scan result, or just your domain. We'll tell you what matters and who should fix it. We'll tell you whether you need a penetration test. There is no obligation, and no pressure to use us for the work.

Book online: meetings.intelamation.net/schedule

Call: (888) 711-4521 **Email:** sales@intelamation.com

Request a penetration test: phantom.intelamation.net

Run the free scan again: intelamation.com/website-security-check.html

Free Microsoft 365 checklist: intelamation.com/m365-security-checklist.html

About Intelligent Automation

Intelligent Automation, LLC is an IT and cybersecurity company in the New York area. We run security for businesses and protect Microsoft 365. We act as a part-time security or technology lead for businesses that don't need a full-time one. These roles are called vCISO and vCTO. We also do penetration testing with Argos Phantom.

Whatever you decide, this guide is yours to use. Nothing in it depends on hiring us. A clean scan or a clean test is proof about what was tested, on the day it was tested. It never promises that a business is safe.

SECTION 13

Questions people ask

> **In plain words**
>
> Short answers to the questions we hear most.

Will the scan slow down or break my site?

It is built not to. The free check sends about 15 requests. The deeper check sends at most 4 a second and 200 in total. It never sends a request body, PUT or DELETE. It skips links and forms that look like actions.

Is it legal to scan my own site? Someone else's?

Scanning a site you own or manage is normal. For anyone else's site, get written permission first. Use email B in section 9. The deeper check won't run until someone proves they control the domain. This is general information, not legal advice.

Why did I get "could not check"?

The site didn't answer. Or a firewall blocked the request. Or a lookup took too long. The finding says which. It never counts as a pass. Too many of them means no grade.

I fixed something, but the scan still shows it.

A free check of the same domain within 10 minutes gives back the earlier result. Wait and scan again. Domain changes can take hours to spread. Your host may also show an old copy for a while.

Does an A mean I'm safe?

No. It means fewer known problems in what we checked, on that day. No outside scan can prove a site is safe.

What do you keep about my scan?

We keep results for about 15 minutes. Then we delete them. To stop misuse, we log each scan. The log keeps the time, your connection's address and the domain. For the deeper check, the log keeps a scrambled code made from your email, not the email itself. If email is switched on, we hold your address for up to 24 hours to send the report.

Why can't I run the deeper check from my IT provider's email?

The email must be at the domain you're proving. The proof must be on that domain's servers or records. Your provider can help you add the file, tag or record. But the email address must be yours.

Our firewall blocked the scan. Should we let it through?

You don't have to. A block shows your protection working. If you want a full grade, you could let the scanner through for a short time. Then remove the rule. Many businesses leave the block in place and accept "could not check".

Do you do penetration tests?

Yes, with Argos Phantom (section 11). Every test is planned and approved in writing first. Request one at phantom.intelamation.net, or call (888) 711-4521.

Is this enough for compliance, insurance or a customer audit?

It can be useful proof that you check your website and act on what you find. But it is not an audit, a penetration test or a certificate. Ask the person who set the rule what they accept.

SECTION 14

Words you will see (glossary)

Active Directory. The system many businesses use to run Windows logins.

Attack surface. Everything a stranger can reach from the internet without your permission.

CAA. A record that names the companies allowed to make certificates for your domain.

Certificate. A file that proves a site is who it says it is. It makes the padlock work.

Cipher. A way of scrambling data. Some old ones (RC4, 3DES) are broken.

Content-Security-Policy (CSP). A setting that tells browsers which code your pages may load.

Cookie. A small note a site leaves in a visitor's browser, such as a login.

CORS. The rule for which other websites may read your pages in a browser.

Cross-site scripting (XSS). When a page shows visitor text as code, so someone else's code can run on your site.

CVE. A public ID number for a known software weakness, such as CVE-2020-11022.

DMARC. A record that tells mail servers what to do with fake email from your domain.

DNS. The internet's address book. It turns names into server addresses and holds email settings.

DNSSEC. A seal that shows your domain records are real.

Encryption. Scrambling data so only the right person can read it.

Firewall. A filter that decides which connections may reach your systems.

Header. A short setting your site sends with each page.

HSTS. A setting that tells browsers to always use the locked address.

HTTP method. The type of a web request, like GET, POST, PUT, DELETE, OPTIONS or TRACE.

MTA-STS. A rule that asks other mail servers to lock email sent to you.

Multi-factor sign-in (MFA). A login that needs a second step, like a code on your phone.

Penetration test. A planned attempt to break in, run by qualified people within agreed limits.

Phishing. A fake email that tries to trick someone into clicking or sharing a password.

Port. A numbered door on a server. Each service listens behind one.

Ransomware. Harmful software that locks your files and asks for money.

Redirect. When a page sends a visitor to another address.

Rules of engagement. The signed document that sets what a test may and may not do.

security.txt. A file that tells researchers how to report a problem to you.

SPF. A record that lists the servers allowed to send email as your domain.

SQL injection. When visitor text gets into a database command and changes it.

Subdomain. An address under your main one, like shop.example.com.

Subdomain takeover. Someone claims a service your old address still points to, and posts under your name.

TLS. The lock behind the padlock. Versions 1.2 and 1.3 are current. 1.0 and 1.1 are retired.

VPN. A private, locked connection into your network for staff.

Weakness (vulnerability). A flaw that could let someone in or do harm.

Web firewall (WAF). A filter in front of a site that blocks harmful requests.

Fix plan worksheet

Print this page. Copy each finding from your report, worst first. Use it when you agree who does what.

DOMAIN _____ SCAN DATE _____ GRADE AND SCORE _____

CHECK RUN ☐ Free check ☐ Deeper check NEXT SCAN DUE _____

#	Finding (title from the report)	Severity	Who fixes it	Target date	Done on	Rescan OK
1	*e.g. No DMARC policy*	*High*	*IT provider*	*[date]*		☐
2						☐
3						☐
4						☐
5						☐
6						☐
7						☐
8						☐
9						☐
10						☐
11						☐
12						☐
13						☐
14						☐

ACCEPTED AS THEY ARE Findings you chose not to fix. Say why, who decided and when. Look again each quarter.

Finding	Why it is acceptable for us	Who decided	Date

One-page checklist

Tick these off as you go. Each line matches a finding or a step in section 8.

This week

- [ ] Renew any certificate that runs out within a month. Turn on automatic renewal.
- [ ] Remove every private file left in public. Then change the passwords and keys inside.
- [ ] Close open database, remote desktop, file sharing and remote control doors (ports).
- [ ] Start email protection (DMARC) in report-only mode, with an address for the reports.
- [ ] Fix your SPF record if it ends in +all, or if you have more than one.
- [ ] Send every critical and high finding to its owner with a date (email A).

This month

- [ ] Send every http visit to https. Set HSTS for at least 6 months.
- [ ] Allow only the current versions of the lock (TLS 1.2 and 1.3). Refuse RC4 and 3DES.
- [ ] Update your site builder, add-ons and code libraries. Remove what you don't use.
- [ ] Open login pages only to your office or a private connection. Or add a second login step.
- [ ] Switch off TRACE, PUT and DELETE, unless something on your site needs them.
- [ ] Fix any link that bounces visitors elsewhere, and any page that trusts visitor text.
- [ ] Turn on Content-Security-Policy in report-only mode.

This quarter

- [ ] Add the other browser settings: framing, nosniff, Referrer-Policy and Permissions-Policy.
- [ ] Set the Secure, HttpOnly and SameSite flags on cookies.
- [ ] Turn email protection (DMARC) up to spam, then to refuse.
- [ ] Publish CAA and security.txt. Set up MTA-STS in testing mode. Turn on DNSSEC.
- [ ] Delete domain records for old services and old web addresses.
- [ ] Write down who owns the site, the hosting, the domain and email.

Every time the site changes

- [ ] Run the free check. Run the deeper check at least once a quarter.
- [ ] Compare the result with the last report. Keep both.

Checked by _____ Date _____