Vulnerability Disclosure Policy Template

A free template for product teams

Sylvan Assurance, LLC

Version 1.0 · Updated 2026-09-03

Contents

Vulnerability Disclosure Policy (VDP) **1**
[Organisation name] . . . 1
Purpose . . . 1
Scope . . . 2
How to submit a report . . . 2
What we commit to . . . 2
Safe harbor . . . 3
Coordinated vulnerability disclosure (CVD) . . . 3
What we ask of you . . . 3
Bug bounty program . . . 4
Contact . . . 4
Version history . . . 4
Drafting guidance (read before customising) . . . 4
Cross-References . . . 10

About This Template **11**

Vulnerability Disclosure Policy (VDP)

[Organisation name]

Effective Date: [YYYY-MM-DD] **Version:** 1.0 **Owner:** [Security Lead / Chief Technology Officer (CTO) / Chief Information Security Officer (CISO)] **Review Cycle:** Annual

---

Purpose

[Organisation Name] is committed to the security of our products and services and the protection of our customers' data. We recognise that security researchers and members of the public play an important role in identifying vulnerabilities before they can be exploited.

This Vulnerability Disclosure Policy describes how to report a security vulnerability to us, what you can expect from us in response, and the protections we extend to researchers

who act in good faith.

Scope

This policy applies to security vulnerabilities in:

In Scope	Out of Scope
[Product Name], [URL or version range]	Third-party services we do not control
[Product Name], [URL or version range]	Physical security (offices, hardware)
[organization-domain.com] and subdomains	Social engineering of our employees
Customer-facing APIs at [api.organization.com]	Denial of service (DoS/DDoS) attacks
Mobile applications: [App Name] (iOS/Android)	Volumetric or automated scanning without prior approval

If you are unsure whether a system is in scope, contact us before testing.

How to submit a report

Submit all vulnerability reports to:

Email: security@[organization.com] **PGP Key:** [Link to public PGP key, if available] **Web Form:** [https://organization.com/security/report] *(if applicable)* **Response commitment:** We will acknowledge receipt within **[2 business days]**.

What to include in your report

To help us triage and respond quickly, please include:

- A clear description of the vulnerability and its potential impact
- The affected product, service, URL, or component
- Step-by-step instructions to reproduce the issue
- Proof-of-concept code, screenshots, or video (if available)
- Your assessment of severity (Critical / High / Medium / Low)
- Whether you believe the vulnerability is currently being exploited
- Your preferred contact method and whether you wish to be credited

You may submit anonymously. Anonymous submissions are accepted but limit our ability to ask clarifying questions or notify you of resolution.

What we commit to

When you submit a report in good faith under this policy, we commit to:

1. **Acknowledge** your report within **[2 business days]** of receipt

2. **Provide an initial triage assessment** within **[5 business days]**, including a severity classification
3. **Keep you informed** of our progress at intervals of no longer than **[14 days]**
4. **Notify you** when the vulnerability has been remediated
5. **Credit you** in any public disclosure, unless you prefer to remain anonymous
6. **Not pursue legal action** against you for security research conducted in compliance with this policy

---

Safe harbor

We will not initiate legal action against researchers who:

- Report a vulnerability in good faith through the channels defined in this policy
- Make a reasonable effort to avoid privacy violations, service disruption, or harm to our users
- Do not access, modify, delete, or exfiltrate data beyond what is necessary to demonstrate the vulnerability
- Do not publicly disclose the vulnerability before we have had a reasonable opportunity to remediate it (**[90 days]** from the date of our acknowledgment, unless otherwise agreed)
- Do not use the vulnerability for personal gain or to harm [Organisation Name] or its customers

If you comply with these conditions, [Organisation Name] will not pursue civil or criminal action related to your research. We consider your research to be conducted in good faith.

Coordinated vulnerability disclosure (CVD)

We follow the coordinated vulnerability disclosure model. This means:

1. You report to us privately
2. We work to remediate the vulnerability
3. After remediation (or after **[90 days]** from acknowledgment, whichever comes first), we support public disclosure
4. If you intend to publish findings, we ask that you coordinate timing with us so that customers can be protected before details are public

We support researchers who wish to present findings at conferences or publish blog posts. Please contact us in advance so we can coordinate.

What we ask of you

- Do not test in a way that degrades service availability for other users
- Do not access or modify data belonging to users other than your own test accounts
- Do not conduct social engineering, phishing, or physical attacks against our team or customers

- Do not demand payment or bug bounty compensation as a condition of disclosure: if we have a bounty program, it is offered voluntarily

---

Bug bounty program

[Select one of the following and delete the other.]

Option A (no bounty program): [Organisation Name] does not currently operate a paid bug bounty program. We offer public recognition for responsible disclosures where the reporter consents.

Option B (bounty program in place): [Organisation Name] operates a paid bug bounty program. Reward amounts are determined at our discretion based on severity, impact, and quality of the report. See [link to bounty program page] for current reward ranges and scope details.

Contact

For questions about this policy, contact: security@[organization.com]

For urgent or critical vulnerabilities, include **[URGENT]** in the subject line.

Version history

Version	Date	Change
1.0	[YYYY-MM-DD]	Initial policy published

This policy is reviewed annually and updated as our product and organisational scope evolves.

Drafting guidance (read before customising)

The template above is the working starting point. The guidance below covers the decisions an organisation makes when customising it to their specific product, customer base, regulatory environment, and risk posture. A VDP that is copy-pasted without these decisions being made deliberately is a VDP that will not survive contact with its first significant incident.

How a VDP differs from a bug bounty programme

A *Vulnerability Disclosure Policy* (VDP) is the public commitment that defines how researchers can report vulnerabilities to you. It does not, by itself, involve payment.

A *bug bounty programme* is a separate (and additional) commercial arrangement where you pay researchers for valid reports.

You can have a VDP without a bounty programme; you cannot have a bounty programme without a VDP. Most organisations should publish a VDP first, then add a bounty programme later (often months or years later) once the Product Security Incident Response Team (PSIRT) is mature enough to handle the volume.

This template is for the VDP. For depth on running a bounty programme, see **5.12 Bug Bounty Operations Manual**.

The scope decision in practice

Scope is the single most consequential customisation. Get it wrong in either direction and the programme suffers.

Too narrow scope invites researcher frustration. Researchers find something genuinely valuable, get told it's out of scope, and either publish publicly or stop engaging. Both outcomes are worse than expanding scope.

Too broad scope invites overwhelming volume. Every kiosk, every internal subdomain, every legacy property becomes fair game. Triage capacity gets consumed by reports against things you may not even still operate.

The recommended pattern:

1. **Start narrow.** Your flagship product or service plus the main marketing site. Nothing else.
2. **Expand deliberately.** As the PSIRT proves it can handle the volume, add the next-most-important property: your customer-facing API, then your mobile app, then your secondary product.
3. **Document the out-of-scope explicitly.** "Third-party services we do not control" is fine; specific named examples ("Marketing automation hosted at vendor.com is out of scope") are better.
4. **Provide a path for ambiguous cases.** Encourage researchers to ask before testing if they are unsure. Acknowledge that you will treat good-faith ambiguity reports with the same safe-harbour as in-scope.

Common scope clarifications worth including:

- *Authentication and authorisation:* "Findings affecting the authentication of our flagship product are in scope. Findings affecting our marketing-site analytics platform's authentication are out of scope; please report those directly to [analytics vendor]."
- *Customer-deployed instances:* For self-hosted or customer-installed products, clarify whether the scope includes the customer's instance or only the vendor-controlled reference installation.

- *Old versions:* If you support multiple versions, specify which versions are in scope. Findings against end-of-life versions are usually out of scope unless they also affect supported versions.
- *Test environments and staging:* Usually out of scope; if you operate a researcher-accessible staging environment, document it explicitly.

Safe-harbour language — why it matters and how to strengthen it

Safe-harbour language is the legal commitment that you will not pursue legal action against researchers who comply with the policy. It is the single most important trust signal in a VDP: researchers in many jurisdictions face real legal risk under computer-misuse laws (CFAA in the US, Computer Misuse Act in the UK, similar in many countries), and a credible safe-harbour reduces that risk materially.

The template's safe-harbour language is a reasonable baseline. To strengthen it:

1. Cite the framework you're modelled on. The most credible safe-harbours reference the DOJ guidance on CFAA (US), the Disclose.io safe-harbour wording, or sector-specific frameworks. Citing these signals legal sophistication and aligns your policy with peer practice.

2. Address the international dimension. A US-based organisation's safe-harbour does not legally extend to acts in other jurisdictions. Consider adding: "Where local laws conflict with this policy, we will work with affected researchers to seek appropriate accommodations and will not initiate legal action where the research was conducted in good faith under this policy."

3. Clarify the "good faith" interpretation. What you mean by good faith is what protects the researcher. Examples that strengthen the commitment: - "We interpret 'good faith' broadly. A researcher who follows the steps in this policy and acts to minimise harm is acting in good faith, even if some incidental data exposure occurred during demonstration." - "Good-faith research does not require perfection. Minor accidental disclosure or service disruption during normal vulnerability investigation does not, by itself, void safe-harbour."

4. Address third-party legal risks. Sometimes a researcher testing your product touches infrastructure operated by your cloud provider, CDN, or other vendors. The safe-harbour should commit to assist: - "Where security research conducted under this policy may have touched systems operated by our third-party providers, we will engage those providers on the researcher's behalf to clarify that the research was authorised."

5. Make the safe-harbour public-facing. Some organisations bury safe-harbour in legal terms; the most credible programmes publish it prominently on the security page. The signal matters as much as the wording.

The disclosure timeline — why 90 days, and when to vary it

The 90-day default in the template aligns with industry practice (Google Project Zero, Microsoft, CERT/CC, ZDI). Researchers know what 90 days means; deviating from it requires justification.

When to extend. Some incidents warrant longer disclosure timelines: - *Complex supply-chain coordination* (multi-vendor disclosure may need 120+ days). - *Hardware vulnerabilities* requiring firmware update cycles (Intel/AMD CPU issues have shipped at 6-12 month embargoes). - *Critical infrastructure* where customer patching cadence is unusually long (some industrial-control system issues run 180+ days).

When to compress. Some findings warrant shorter: - *Active exploitation in the wild*: disclosure should happen ASAP with available mitigation guidance. - *Already-public information*: there's no embargo to honour if details are already in trade press.

The negotiation language. The template says "unless otherwise agreed." That phrase is doing important work. It acknowledges that the default is a default and that variations are normal. Researchers respect a policy that says "we want to do this in 90 days but we'll discuss if your finding warrants something different."

Coordinated vulnerability disclosure (CVD) — the framework you're adopting

The template references CVD without explaining it. For credibility, you may want to expand:

- *What CVD is.* The framework where researcher and vendor coordinate on the timing and content of public disclosure, allowing the vendor to develop and ship a fix before details become public.
- *Why it exists.* Without coordination, researchers either sit on findings indefinitely (bad, exploitation may already be happening) or publish immediately (bad, customers have no patch yet). CVD is the working compromise.
- *Where it's documented.* International Organization for Standardization (ISO)/IEC 29147 (Vulnerability disclosure), ISO/IEC 30111 (Vulnerability handling processes), the Forum of Incident Response and Security Teams (FIRST) PSIRT Services Framework, and various industry bodies (CERT/CC, ICASI, etc.).
- *What good engagement looks like.* The expanded CVD playbook in **5.1** provides the deep treatment.

A VDP that references and explains CVD signals more sophistication than one that just mentions it in passing. Consider adding: *"For background on CVD, see ISO/IEC 29147 or the FIRST PSIRT Services Framework."*

Jurisdiction-Specific Considerations

The template is jurisdiction-agnostic. Specific jurisdictions have specific expectations:

United States. - *CFAA exposure for researchers.* The Computer Fraud and Abuse Act has historically been used aggressively against security researchers. The 2022 DOJ policy change explicitly protects "good faith" research, but the statute remains. A strong safe-harbour matters. - *DMCA Section 1201.* The Digital Millennium Copyright Act's anti-circumvention provisions have been used against researchers studying DRM. Triennial exemptions exist; reference if your product involves DRM or copy protection. - *"Establishing a Coordinated Vulnerability Disclosure Program to Work With Security Researchers"*, the July 2026 joint guidance from the Cybersecurity and Infrastructure Security Agency (CISA), the National Security Agency (NSA), Japan's JPCERT/CC, the

Netherlands' NCSC-NL, and the United Kingdom's NCSC-UK, and the CISA VDP Platform are useful reference points.

European Union. - *EU Directive on the resilience of critical entities and NIS2* both reference vulnerability disclosure expectations. - *EU Cyber Resilience Act* (CRA) makes coordinated vulnerability disclosure a regulatory requirement for products with digital elements placed on the EU market. Your VDP must comply with CRA's requirements (see **5.7 §6**). - *General Data Protection Regulation (GDPR) consideration.* Vulnerability disclosure may touch personal data; the policy should clarify that researchers must minimise data access and report rather than retain.

United Kingdom. - *Computer Misuse Act 1990.* Similar to CFAA in spirit; aggressive interpretation has been used against researchers historically. Strong safe-harbour matters. - *NCSC Vulnerability Disclosure Toolkit* provides UK-specific guidance. - *UK PSTI (Product Security and Telecommunications Infrastructure Act).* For connected consumer products, requires a vulnerability disclosure mechanism. If your products are in scope, the VDP must comply with PSTI requirements.

Other jurisdictions. Brazil, India, Singapore, Japan, Australia each have evolving computer-misuse frameworks. Counsel familiar with the jurisdictions you operate in should review the VDP.

Bug bounty programme integration

The template includes two options for the bug bounty section. Some additional patterns:

Public VDP + private bounty programme. Common pattern. The VDP is open to anyone; a bounty programme operates separately on invitation. The VDP says: "We offer public recognition; researchers may be invited to our private bug bounty programme based on the quality and impact of their contributions."

Public VDP + public bounty programme. Common pattern at maturity. Both are publicly documented; the bounty programme has its own page with reward ranges and scope.

Tiered approach. Public VDP (no bounty) for the broad attack surface; private bounty programme for high-value targets. Some researchers contribute to both; the boundary is documented in the bounty programme's terms.

For depth on building and running the bounty programme, see **5.12 Bug Bounty Operations Manual**.

Common VDP failure modes

Patterns that cause programmes to underperform:

1. The unreplied report. A researcher submits via `security@`; nobody monitors the inbox; the researcher waits a week and then goes public. The VDP is only credible if the channel is actively monitored. Test it.

2. The unstated safe-harbour. The policy is silent on legal protection; researchers assume there is none. Even good safe-harbour language gets ignored if it isn't prominent.

3. The narrow safe-harbour. Safe-harbour conditional on so many things that researchers cannot rely on it. Common over-restriction: "must obtain prior written approval before testing" defeats the point of a VDP.

4. The unmaintained policy. The policy is dated 2019; researchers wonder if it's still in effect; trust degrades. Annual review and version-bumping (even with no substantive changes) signals active stewardship.

5. The bounty-bait-and-switch. The VDP implies a bounty is available; the researcher submits; the bounty is denied. If a bounty is offered, it's part of the bounty programme's terms; the VDP itself should not promise rewards.

6. The credit dispute. Researcher requests credit; vendor publishes without credit; researcher complains publicly. Always confirm credit wording with the researcher before publication. See **5.8 T-11**.

7. The expanded scope without expanded capacity. Programme grows; report volume exceeds triage capacity; SLAs slip; researcher reputation drops. Match scope to capacity.

8. The vague disclosure timeline. "We will work with you on disclosure timing" without a default. Researchers interpret this as "we will delay indefinitely." Specify a default (90 days is standard) and the conditions for variation.

Adoption checklist

Before publishing the policy:

- ☐ Scope reviewed and tightened to current capacity.
- ☐ `security@` inbox actively monitored with on-call coverage.
- ☐ Safe-harbour language reviewed by legal counsel familiar with the relevant jurisdictions.
- ☐ PGP key generated and published (or alternative secure-submission path).
- ☐ Acknowledgement template (from 4.2 Intake Form or 2.1 Playbook) ready to send.
- ☐ Triage workflow documented (handoff to engineering, severity scoring, customer-comms integration).
- ☐ Bounty section reflects current state (no programme vs programme running).
- ☐ Internal stakeholders briefed: legal, engineering leadership, executive sponsor.
- ☐ Policy published at a stable URL (often `/security`, `/security/disclosure`, or `/.well-known/security.txt`).
- ☐ `security.txt` file published at `/.well-known/security.txt` per RFC 9116, pointing to the policy.
- ☐ First-year review date scheduled.

After publishing:

- ☐ Acknowledge first inbound report within stated Service-Level Agreement (SLA) (this is the credibility test).
- ☐ Track metrics: report volume, valid-report rate, time-to-acknowledge, time-to-resolution.
- ☐ Annual review held; any updates published as v1.1 etc.
- ☐ Customer notification of the VDP via product communications and partner channels.

Cross-References

- **5.1 Coordinated Vulnerability Disclosure In-Depth Playbook**: for the full CVD framework this policy operationalises.
- **5.2 Working with Security Researchers**: for the researcher-relationship discipline.
- **5.3 §5**: process foundations including VDP placement in the PSIRT function.
- **5.7**: regulatory dimensions of disclosure obligations.
- **5.12 Bug Bounty Operations Manual**: when and how to add a paid programme.
- **4.2 Responsible Disclosure Intake Form**: the structured report submission template.

A VDP is a public commitment. It signals to the security community that you welcome research and that you can be engaged constructively. The policy itself is necessary but not sufficient. What makes the VDP credible is how it is operated. Publish a policy you can actually honour, then honour it.

About This Template

This template is produced by **Sylvan Assurance, LLC** and reflects more than twenty-six years of professional security experience, including the most recent eight years in product security incident response and vulnerability disclosure.

It provides **general guidance and a recommended starting point**. It is **not** legal advice. A Vulnerability Disclosure Policy (VDP) is a public-facing legal document. Have it reviewed by qualified legal counsel before you publish it. Responsibility for your business's security and for the content you publish remains with you.

For full terms, see the Terms of Use provided with PSIRT Response.

Copyright 2026 Sylvan Assurance, LLC. Not for resale.

Free to share unchanged with credit. Terms: sylvanassurance.com/licence/free/

Where to go from here



Scan for the full toolkit that builds on this guide.
sylvanassurance.com/go/toolkit-psirt-response



Scan for the companion book - the full story, in plain English.
sylvanassurance.com/go/psirt