

Writing an Incident Response Plan Before You Need One

Wed, Sep 9, 6:00 PM EDT · <https://scottslab.io/posts/incident-response-plan>



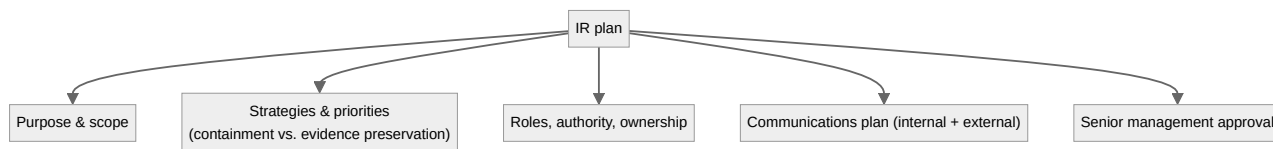
TL;DR

An IR plan exists so you make good decisions under pressure by having made them in advance, calmly. It spells out purpose and scope, strategies and priorities (the big one: containment vs. evidence preservation), roles and authority, a communications plan, and senior management approval. That approval is what lets you take unpopular actions mid-incident. NIST SP 800-61 (Revision 3, 2025) is the reference, and it isn't the only lifecycle model worth knowing: SANS's six-step version slices the same ground more finely, and this series follows its steps.

The point of an incident response plan isn't the document: it's that you make the hard calls when you're calm, so you're not making them at 3am with an attacker in the network and executives on the phone. Structure prevents bad decisions under pressure. The worst incident-response mistakes I've seen weren't technical; they were someone improvising a decision the plan should have already made.

A usable plan has a handful of elements. A statement of purpose and scope: what this plan covers and when it applies. Strategies and priorities, and the one that matters most is stated explicitly: containment versus evidence preservation. Those two goals conflict: pulling the plug contains the

damage but destroys volatile evidence. So the plan should say, in advance, which one wins for which kind of incident, because that's not a debate you want to have live. Roles, authority, and ownership, so everyone knows who decides what. A communications plan for both internal and external audiences. And senior management approval. That sounds like a formality and is actually the most important line in the document.

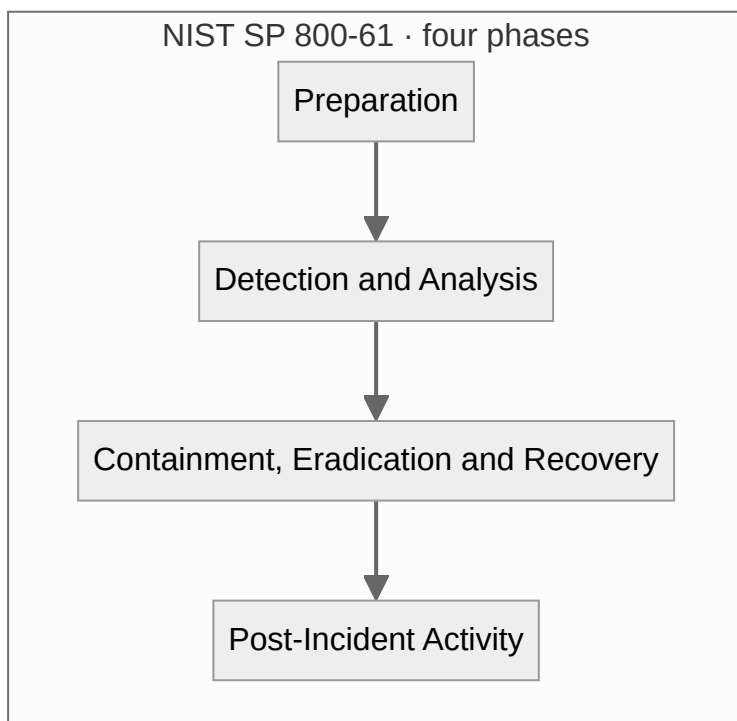
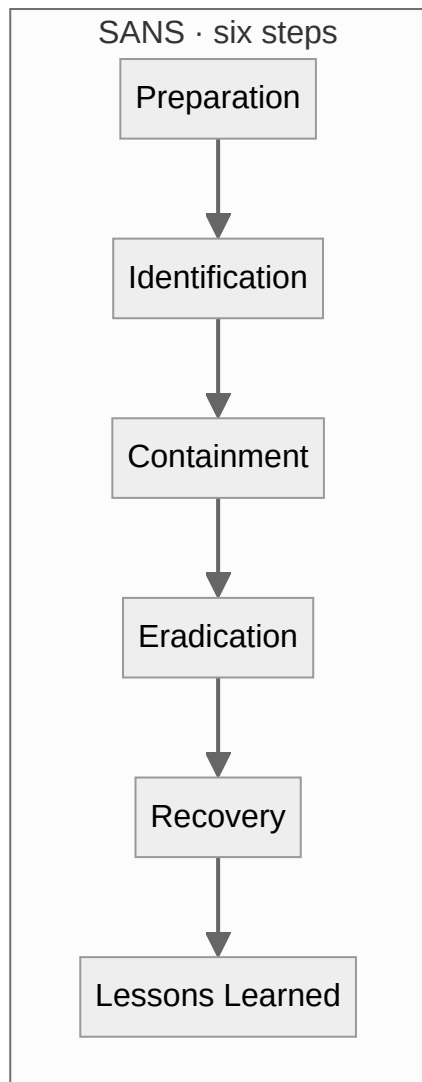


Writing an Incident Response Plan Before You Need One

That last element deserves its own paragraph. Incident response routinely requires unpopular actions: taking a revenue-generating system offline, cutting off a business unit's access, calling in outside help that costs money. When those calls come mid-incident, the responder needs the authority to make them stick, and that authority comes from a plan the executives signed *before* anyone was under fire. Without it, every hard decision turns into a negotiation with the person whose system you're about to unplug, and you lose the minutes that matter.

Two lifecycles, same shape

NIST SP 800-61 is the canonical reference for the whole framework this series follows. Its 2025 Revision 3 (titled *Incident Response Recommendations and Considerations for Cybersecurity Risk Management: A CSF 2.0 Community Profile*) reframes incident response around the six functions of the Cybersecurity Framework 2.0, where the earlier Revision 2 (the 2012 *Computer Security Incident Handling Guide*) laid out a simpler four-phase lifecycle: Preparation; Detection and Analysis; Containment, Eradication, and Recovery; and Post-Incident Activity. Rev 3 doesn't discard that lifecycle so much as translate it. Preparation maps to Govern, Identify and Protect. Detection and Analysis maps to Detect, plus Identify's improvement work. Containment, Eradication and Recovery maps to Respond and Recover, again with a slice of Identify's improvement category. Post-Incident Activity folds back into Identify's improvement category. If your organization already runs its risk program on CSF 2.0, that's a genuine convenience. Incident response stops being a separate discipline bolted onto the side of security and becomes the same vocabulary you already use for everything else.



Two lifecycles, same shape

SANS teaches a six-step version of the same idea, and it's the one this series actually follows post by post: Preparation, Identification, Containment, Eradication, Recovery, Lessons Learned. The difference from NIST's four phases isn't philosophical: SANS just splits "Detection and Analysis" into a distinct identification step, and splits "Containment, Eradication, and Recovery" back out into three separate steps, because in practice those are three different postures, often owned by three different people paying attention to three different things. Know both names. Whoever hands you an incident-response requirement in an audit, a client questionnaire, or a cyber-insurance application will use one framing or the other, and they aren't always careful about which.

What actually has to be written down

NIST's own list of required plan elements is a little different from the "purpose, strategy, roles, comms" shape most teams reach for by instinct: mission; strategies and goals; senior management approval; the organization's structural approach to incident response; how the team communicates internally and with outside parties; metrics for measuring the capability; a roadmap for maturing it; and how the whole thing fits into the wider organization. Two items on that list are easy to skip and shouldn't be. The roadmap, because a plan that never improves is the one that fails the same way twice. And the instruction to review the plan at least annually. That's a specific, explicit requirement in the guidance, not a suggestion, because a plan nobody has reopened since it was written is closer to folklore than to a control.

One layer down from the plan sits something people conflate with it and shouldn't: procedures. The plan is policy: what the organization has decided and who's allowed to decide it. Procedures are the standard operating checklists built on top of that policy: the specific technical steps, forms, and decision points a handler actually works through during an incident. NIST is direct about why that layer exists separately: following a standardized, rehearsed procedure minimizes the errors that stressful, high-pressure situations otherwise produce. A plan tells you who can order a system offline. A procedure tells the analyst exactly which commands to run and which log to pull first once that order is given. Skip the procedure layer and you've written a constitution with no laws under it. Every incident still has to improvise the actual work.

You don't have to write either from scratch. NIST SP 800-61 Rev 3 is free, current, and built around the framework most large organizations already use for everything else in security. SANS's *Incident Handler's Handbook* is the practical companion to it: less framework, more "here is what a handler actually does at 2am." Start from one of those rather than a blank page. The best IR plan is the one that exists, is approved, has procedures underneath it, and has been read before the incident. It is not the perfect one you were going to write someday.

