How to Run a Salesforce Access Review Without Losing a Day

A step by step process for a review that actually covers your user base.

> Short answer: scope the review to a specific trigger, define what correct access looks like before checking anyone, check every user against that definition rather than sampling, and record the specific rule behind every finding. The step that normally takes a day is reconstructing each person's effective access by hand; removing that step is what makes the rest realistic on a schedule.

Why do access reviews get sampled instead of completed?

An access review, done properly, confirms that every user's actual access matches what their role requires. Done partially, it checks a handful of users chosen because they seem likely to be problems, and extrapolates. The reason reviews default to the partial version is cost, not laziness.

Step 1: define the trigger and the scope

A review without a defined trigger tends not to happen at all. Reviews that actually run tend to be tied to a scheduled cadence, a compliance requirement, a specific incident, or a merger or reorg.

Step 2: define what correct access looks like, before checking anyone

This step gets skipped more often than it should. For each role in scope, write down, even informally, what access that role should have. Without this, every finding becomes a judgement call made in the moment.

Step 3: pull the list of users in scope

Pull the actual list from Salesforce, by profile, permission set assignment, or role, depending on how scope was defined.

Step 4: resolve full effective access

For each user: their profile's object and field permissions, every permission set and group assigned individually, role hierarchy position, sharing rules, and manual shares. All of it, for every object in scope. This is the step that costs the most time under a manual process.

Step 5: compare against the definition, and log the outcome

Three outcomes: matches, excess access, or missing access. Log all three, not just the problems, so the review can later prove it was comprehensive rather than a search that stopped at the first few issues.

Step 6: trace excess access to the specific granting rule before fixing

Do not fix based on the symptom. Trace whether excess access comes from the profile, a specific permission set, or a group, because the fix differs materially.

Step 7: record the review itself

Keep a record of when it ran, what scope, who conducted it, and the outcome for every user, including the ones with no finding.

What changes if step 4 is not the bottleneck?

Every other step is judgement, and none of it should be automated away. Step 4 is mechanical: correctly reading Salesforce's own permission metadata and presenting it clearly. Who Sees What performs it in seconds rather than the manual thirty-plus minutes, which determines whether steps 1 through 3 and 5 through 7 happen for your whole user base or for a sample.

Get Who Sees What free on the AppExchange →

Who Sees What is built and maintained by TwinStack Solutions, a Salesforce partner. Questions about setup: info@twinstack.net