

The immutable copy test

Three checks for any locked copy

A locked copy is only as good as the last time someone brought data back from it. These three checks show whether yours would do its job: when it was last used for a restore, who can still remove it or shorten its lock, and how long one restore from it takes. A blank box is a finding.

A locked copy, in plain words

What it is.
A backup copy with a lock on it, also called immutable. Until the lock runs out, nobody can change or delete it through the normal screens.

Where it lives.
A locked snapshot on the storage array, a backup with a lock setting turned on in object storage, or a copy kept offline, such as tape on a shelf.

How long it lasts.
As long as its retention setting says. Shorten that number and the protection shortens with it.

Who holds the key.
Designs differ: a separate security role, two people approving together, a call to the maker's support, or nobody until the date passes. Check 2 finds out which one yours uses.

The three checks

1 The last restore from it

The date someone last brought data back from the locked copy itself, not from the everyday backup, and what was restored. No record counts as never.

What you found	The date, and where it is written down

2 Who could delete or shorten it

Every person or role that can delete the copy, shorten its retention, or release the lock, and whether any of them is an everyday sign-in.

What you found	The date, and where it is written down

3 One timed restore

Restore one real system from the locked copy into a separate test area. Write when you started, when it was usable, and who confirmed it works.

What you found	The date, and where it is written down

> When did anyone last restore from your locked copy, and how long did it take?

How to run it, and what it does not prove

Run the checks in order. Check 1 and check 2 need only logs and settings. Check 3 needs a quiet afternoon and a test area that cannot touch the live system. Write the date on every answer, and run the page again next quarter.

The last restore from it
Restore logs in the backup software, or the array's event log, filtered to the locked copy. Ask who ran it and what they restored. A restore from the everyday backup does not count here.

Who could delete or shorten it
The user and role list on the array or backup software, who can change the retention setting, and the maker's process for releasing a lock. Then ask one question: could one person do it alone?

One timed restore
Pick a system the business would miss first. Start the clock when the restore is asked for, not when the copy starts moving. Stop it when the system's owner says it works, not when the copy finishes.

> If the person in check 2 left tomorrow, who else could release the lock?

What this does not prove

That every copy is good. One clean restore proves that copy, on that day. Older and newer copies are untested until you test them.

That the copy is clean. A copy made after a problem started carries the problem too. Check the restored system before you trust it.

That the top role is covered. Check 2 shows who could remove the copy. It does not show what would stop that person's sign-in from being misused.

That everything comes back at once. One timed restore is one system. Several at the same time can take longer, and the order matters.

Made by StepKnock, a small team in San Diego, paid through our partners when a customer buys. This page sells nothing.

Nothing to sign up for. No cookies, no analytics, and nothing you type here leaves this page.