



NxSAM Container Security — Quick Guide

Quick reference guide for the Container Security module

Product release	**NxSAM 2026.09**
Verified against code on	**2026-09-10**
Document revision	**b441b08-dirty**
Audience	**End users**
Document owner	**eSec Forte Technologies — Product Documentation**

Screenshots are taken from a demonstration data set. Figures are illustrative.

eSec Forte Technologies

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Table of Contents

Container Security

- Overview
- Reading the Container Security screens
- Container Security: every screen, column and action
- Container Security: find the right screen

Reference

- Glossary

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Container Security

Container Security

Container Security covers weaknesses in container images — the packaged applications your platforms run. Findings arrive from container scanning tools such as Prisma Cloud.

How containers differ from other assets

Two differences change how you work with them:

An image is immutable. You do not patch a running container; you build a new image and redeploy. So remediation means rebuilding, and a finding persists until the new image is scanned and confirmed.

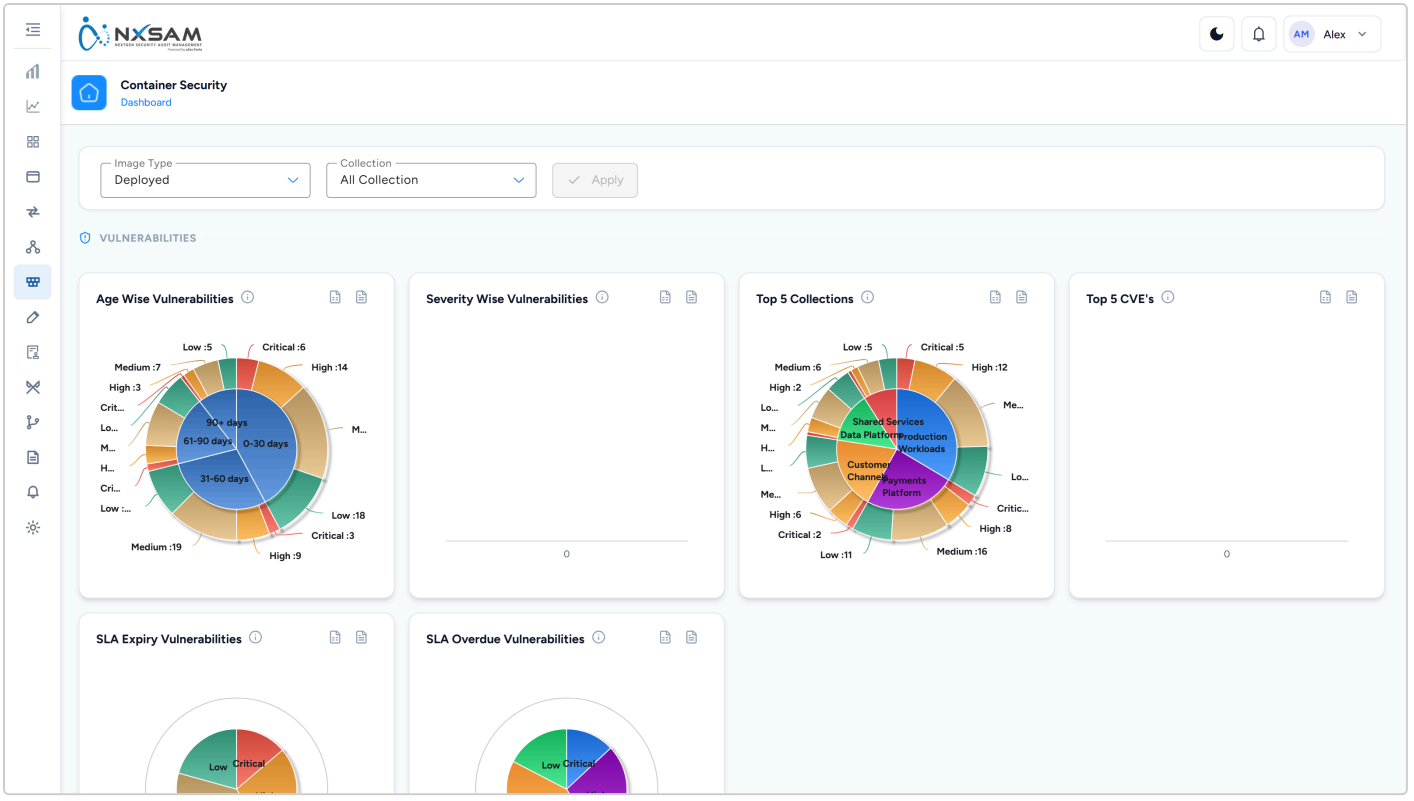
Most findings come from the base image. A container inherits everything in the image it was built from. That means a single base image update frequently resolves large numbers of findings across many images at once — and conversely, an outdated base image reproduces the same findings everywhere it is used.

Fix the base image, not the finding

When the same vulnerability appears across many images, it is usually one inherited component. Updating the shared base image resolves it everywhere. Treating each image separately multiplies the work for no additional benefit.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

What the module holds



The Container Security dashboard.

Screen	Shows
Dashboard	Container posture at a glance
Collections	Logical groupings of images, usually mirroring how your platform organises workloads
Images	The image inventory, by repository and tag
Vulnerabilities	Weaknesses found in those images
Compliance	Checks against container hardening benchmarks

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Images and tags

An image is identified by its repository and tag. The tag matters: `payments/api:2.14.0` and `payments/api:2.15.0` are different images with different findings.

This is why a fix appears as a *new image* rather than as the old one improving. The previous tag still contains the vulnerability — it is simply no longer the one you deploy.

Vulnerabilities versus compliance

The module reports two different kinds of problem, and they are worked differently:

- **Vulnerabilities** are known weaknesses in packages inside the image. Fixed by updating those packages and rebuilding.
- **Compliance** findings are configuration and hardening issues — running as root, no health check, an over-permissive setting. Fixed by changing how the image is built or run.

Both matter, but compliance findings are usually cheaper to fix and often apply uniformly across your estate.

Working effectively

- Start with **collections** rather than individual images — they group workloads the way your platform teams think about them.
- Prioritise images actually deployed to production. A scanned image nobody runs is a lower priority than a running one.
- Watch for the same finding across many images: that is a shared base-image problem and a single high-leverage fix.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Reading the Container Security screens

Container findings behave unlike anything else in NxSAM, because the thing they are attached to is immutable. Most of the confusion in this module traces back to that one fact.

A finding is attached to an image, and an image never changes

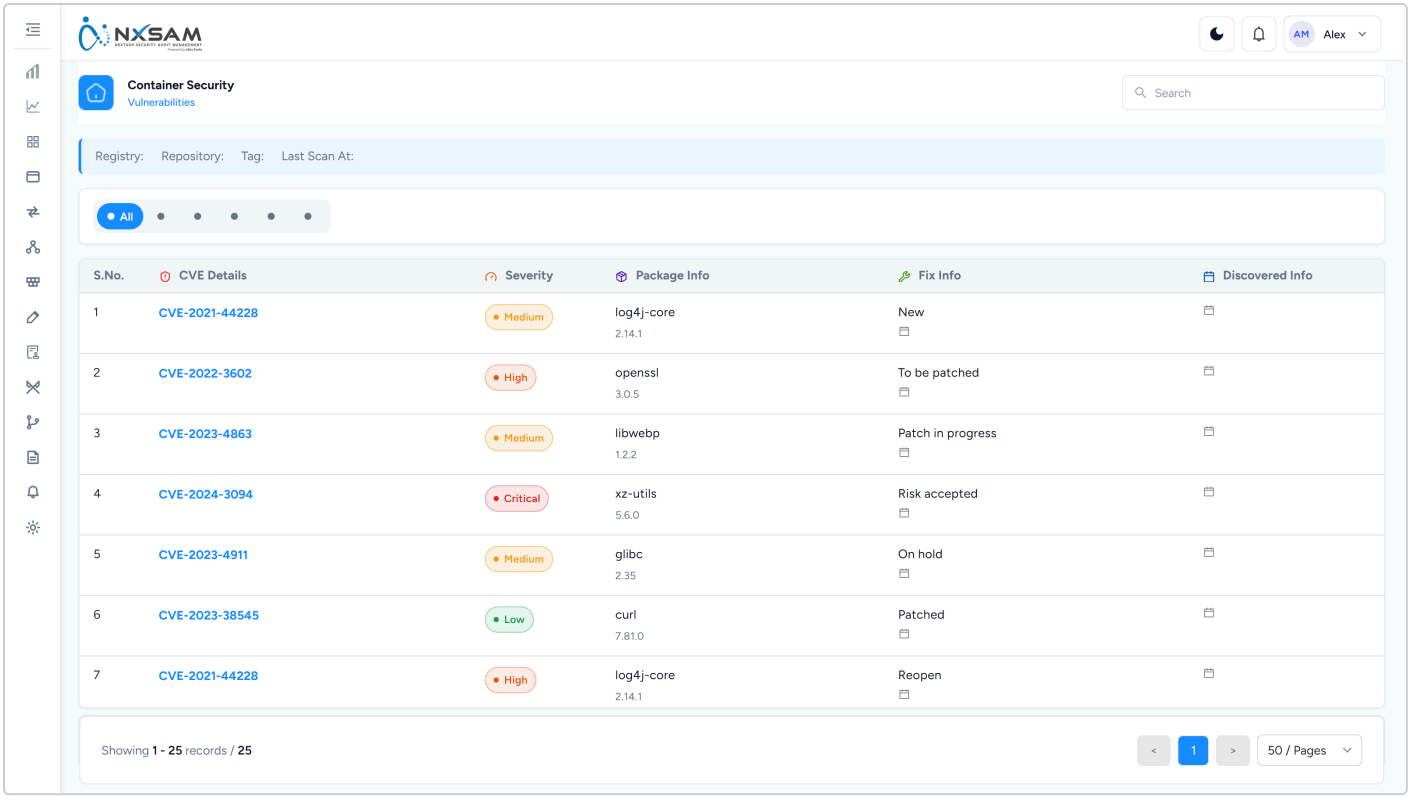
`payments/api:2.14.0` is a fixed artefact. You cannot patch it — you build `payments/api:2.15.0` and deploy that instead.

So:

- A finding on `2.14.0` stays open on `2.14.0` forever. It is not "unfixed"; the image it describes still contains the vulnerability, and always will.
- The fix appears as a **new image with fewer findings**, not as the old one improving.
- "Why is this still open after we fixed it?" is answered by checking which tag is deployed, not by checking the finding.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

The vulnerability columns



Package-level findings inside images.

Column	What it tells you
CVE	The advisory. Shared across every image containing the package
Registry / repository / tag	Which image. The tag is part of the identity
Package name and version	The actual cause. This is what you change to fix it
Collection	The workload grouping the image belongs to
Severity	From the scanner's advisory data

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Column	What it tells you
Vulnerability status	The working state of this finding
Master vulnerability status	The state of the underlying advisory across the estate
Vulnerability age	How long this has been known about
Discovered at / SLA expiry	When it was first seen here, and when it is due

Two status columns again, meaning something different here

Status is this finding on this image. **Master Status** is the advisory itself, across every image it affects.

They diverge legitimately: a CVE can be triaged once at the master level — say, judged not applicable to your base images — while individual findings remain open until the images are rebuilt.

Package name is the column that matters

Severity ranks findings; **package name groups them**. Twenty Critical findings across twelve images are usually three packages, and updating three packages in the base image clears all twenty.

Sorting by package rather than severity is the single most useful habit in this module, and it is why the search screens exist:

- **Container vulnerability search** — findings across every image at once
- **Container compliance search** — the same for configuration checks

Both exist precisely because the per-image view hides the pattern.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Compliance is a different kind of finding

Vulnerabilities are weaknesses in packages inside the image. Compliance findings are how the image is built or run — running as root, no health check, a writable root filesystem, an over-permissive capability.

	Fixed by
Vulnerability	Updating a package and rebuilding
Compliance	Changing the Dockerfile or the runtime configuration

Compliance findings are usually cheaper to fix and apply uniformly, which makes them a good first pass on a new estate — one Dockerfile change can clear the same finding everywhere.

Collections

Images are grouped into collections, normally mirroring how the platform team organises workloads. Start here rather than with individual images: collections match how the people who will do the work think about their systems, and a finding assigned by collection reaches an owner who recognises it.

Why the counts move without a new scan

Same three causes as everywhere else, in the same order: SLA states changing with the date, a severity mapping edit re-classifying findings in bulk, or an image moving between collections and so between report scopes.

The container-specific fourth: **a new tag was scanned**. The old tag's findings did not go anywhere; a new set arrived alongside them. Totals rise even though the estate improved, because you are now tracking two images where you tracked one.

Retiring old tags is what stops that number growing forever, and nothing in the platform does it for you.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Container Security: every screen, column and action

Complements Reading the Container Security screens. This page names every column, including the secret-detection fields that are easy to miss.

Collections

Column	What it holds
Collection Name / Description	The grouping, usually mirroring how the platform team organises workloads
Active Image Count	Images currently in it
Status	Whether the collection is in use
Owner	Who is responsible for the workloads in it

Owner is what makes a container finding assignable. An image has no natural owner — it is built by a pipeline — so the collection carries the accountability.

Images

Column	What it holds
Registry / Repository	Where the image lives
Tag	The version. Part of the identity, not a label on it
Vulnerability	Open package findings

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Column	What it holds
Compliance	Open configuration findings
Last Scan Date	When the image was last evaluated
Collections	Which groupings it belongs to — an image can be in several
Status	Whether the image is active

Row actions — four

Action	What it does
View Vulnerabilities	Package findings for this image
View Compliance	Configuration findings for this image
Export Vulnerabilities	Downloads the vulnerability list
Export Compliance	Downloads the compliance list

Vulnerabilities

Column	What it holds
Registry / Repository	The image
CVE	The advisory

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Column	What it holds
Severity	As the advisory rates it. The column header reads *Serverity* on both container tables — a spelling error in the product, not a different field (FINDINGS #16)
Package Info	The package and version that carries it — the actual cause
Fix Info	Whether a fixed version exists, and which
Discovered Info	When it was first seen here
SLA Overdue Status	Whether it has passed its deadline
Collection	The workload grouping, and therefore the owner
Status	The working state of this finding

Fix Info decides whether a finding is actionable at all

A vulnerability with no fixed version cannot be resolved by rebuilding — the maintainer has not released one. Those findings need a compensating control or a documented acceptance, and treating them as a remediation backlog produces a queue that never moves.

Sorting by Fix Info separates "rebuild will fix this" from "nothing you build will".

Compliance

Configuration and hardening checks, plus secret detection.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Column	What it holds
Registry / Repository	The image
Compliance Details	The check that failed
Category	What kind of check — hardening, configuration, secrets
Severity	As rated
Secret Path	The file a detected secret was found in
Secret Location In File	Where within that file
Discovered Info	When first seen
SLA Overdue Status	Past deadline or not
Collection	Workload grouping and owner
Status	Working state

⚠ Secret Path and Secret Location are themselves sensitive

When a compliance finding is a detected credential, these two columns point directly at it. An exported compliance list therefore contains a map to live secrets in your images.

Treat those exports as sensitive material, and rotate the credential rather than only removing it from the image — it is in every previously built layer regardless.

Page-level exports

- **Export Overall Vulnerabilities** — every finding across every image
- **Export Unique Vulnerabilities** — each CVE once, however many images carry it
- **Export Overall Compliance** — every compliance finding

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

The unique export is the one to use for remediation planning. The overall export counts the same log4j advisory forty times across forty images; the unique export lists it once, which is the number of decisions that actually have to be made.

Per-image detail views

Opening an image shows its findings with a slightly different column set:

Vulnerabilities — `CVE Details`, `Severity`, `Package Info`, `Fix Status`, `Discovered Info`.

Compliance — `Compliance Details`, `Category`, `Severity`, `Title`, `Secret Path`, `Secret Location In File`, `Created At`.

Which images carry a finding

From a vulnerability or compliance finding you can see every image affected:

Column	What it holds
Tag	The image version
First Scan At / Last Scan At	The window over which it has been present
Vulnerability Count / **Compliance Count**	How many findings that tag carries

This is the view that shows a base-image problem for what it is. A CVE appearing against fourteen tags is one package to update, not fourteen tickets — and the first and last scan dates show whether newer tags have already resolved it.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

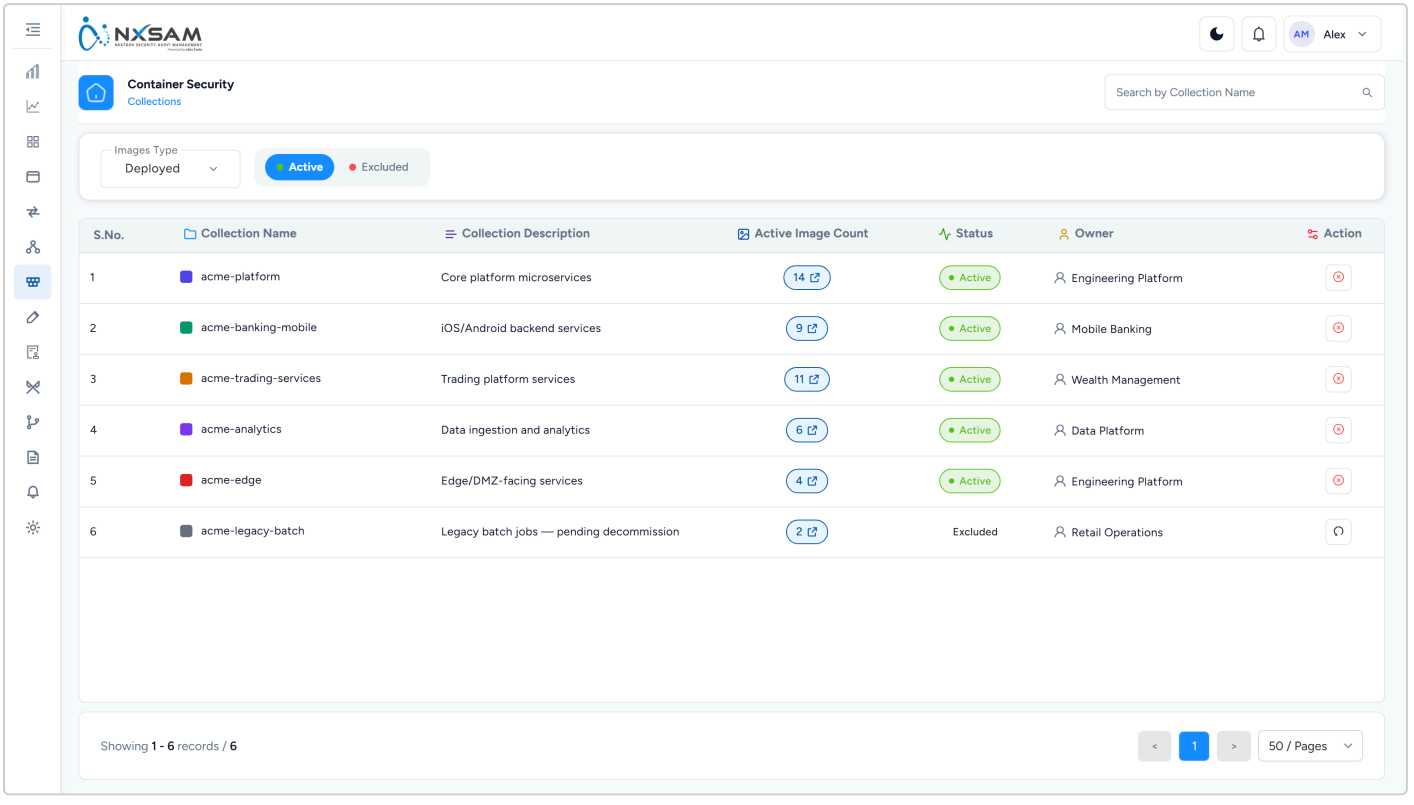
Container Security: find the right screen

Every screen in this module, what each one is for, and where it lives. Use it to find the right screen quickly, and to confirm you are on the one a colleague means — several screens in this product have similar names and different jobs.

Screenshots come from a demonstration data set, so the figures are illustrative. Click any screenshot to open it at full resolution.

Container Collections

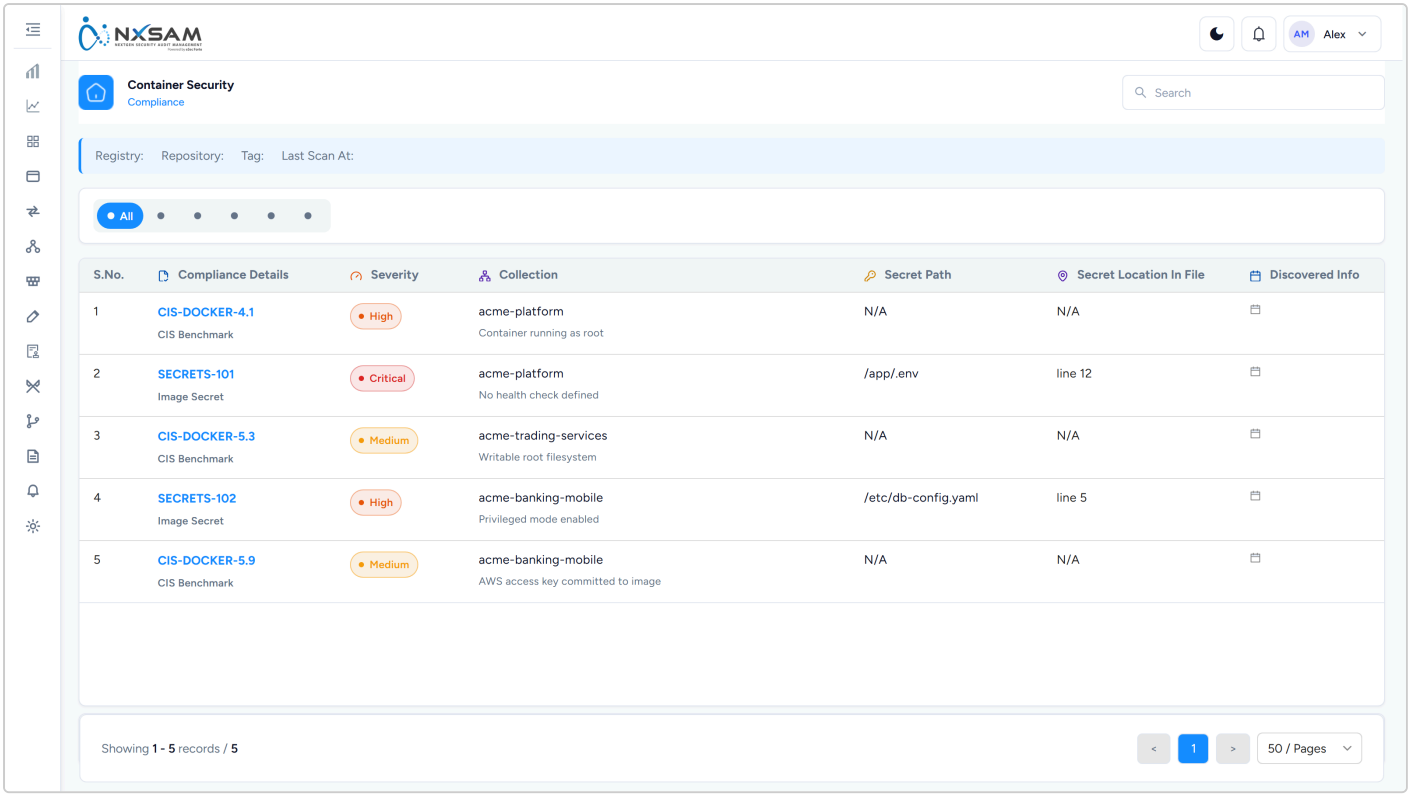
Logical groupings of images, usually mirroring how the platform team organises workloads.



Container Security → Collections

Container Compliance

Hardening and configuration checks against images, as distinct from package vulnerabilities.

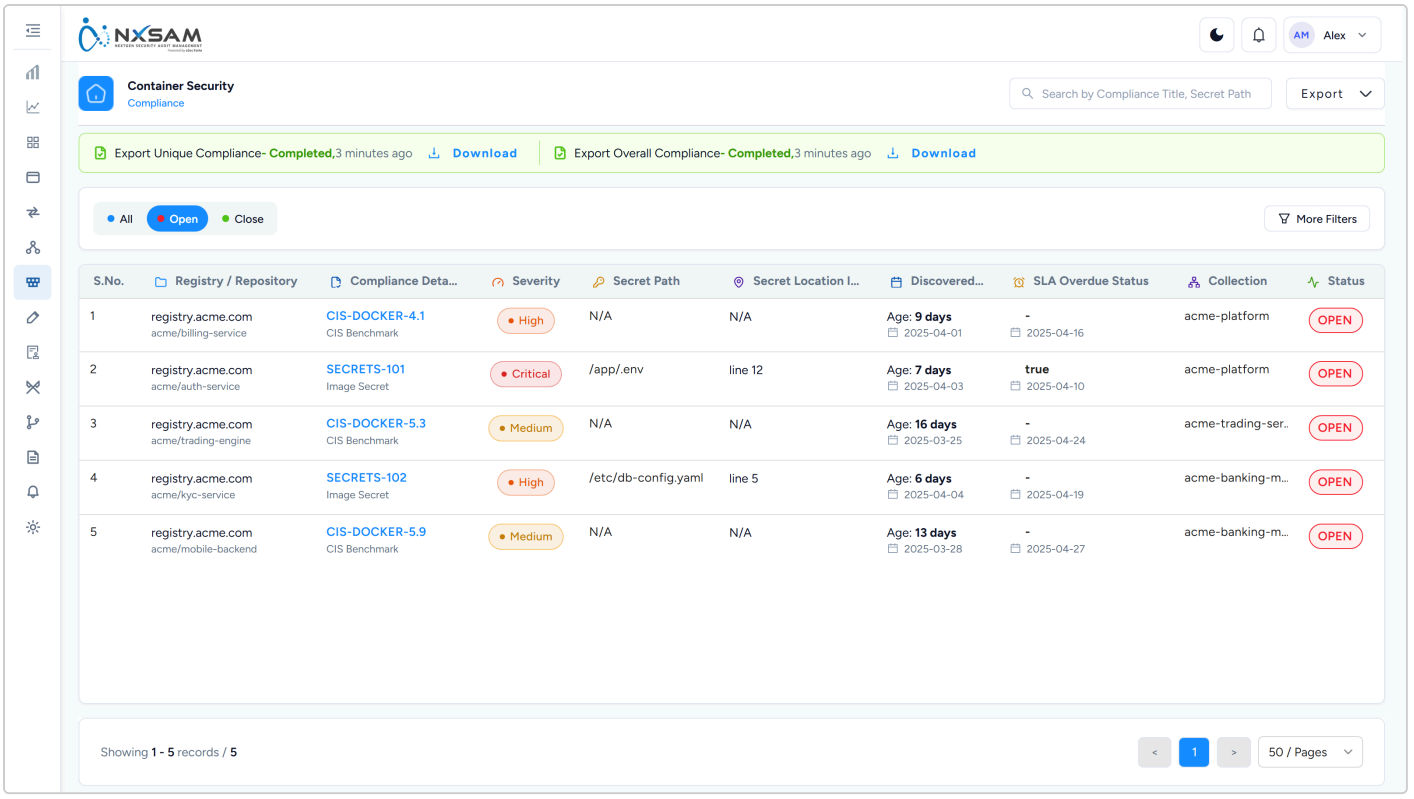


Container → Container Compliance

Container Compliance Search

Search compliance findings across all images, for spotting a setting wrong everywhere.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

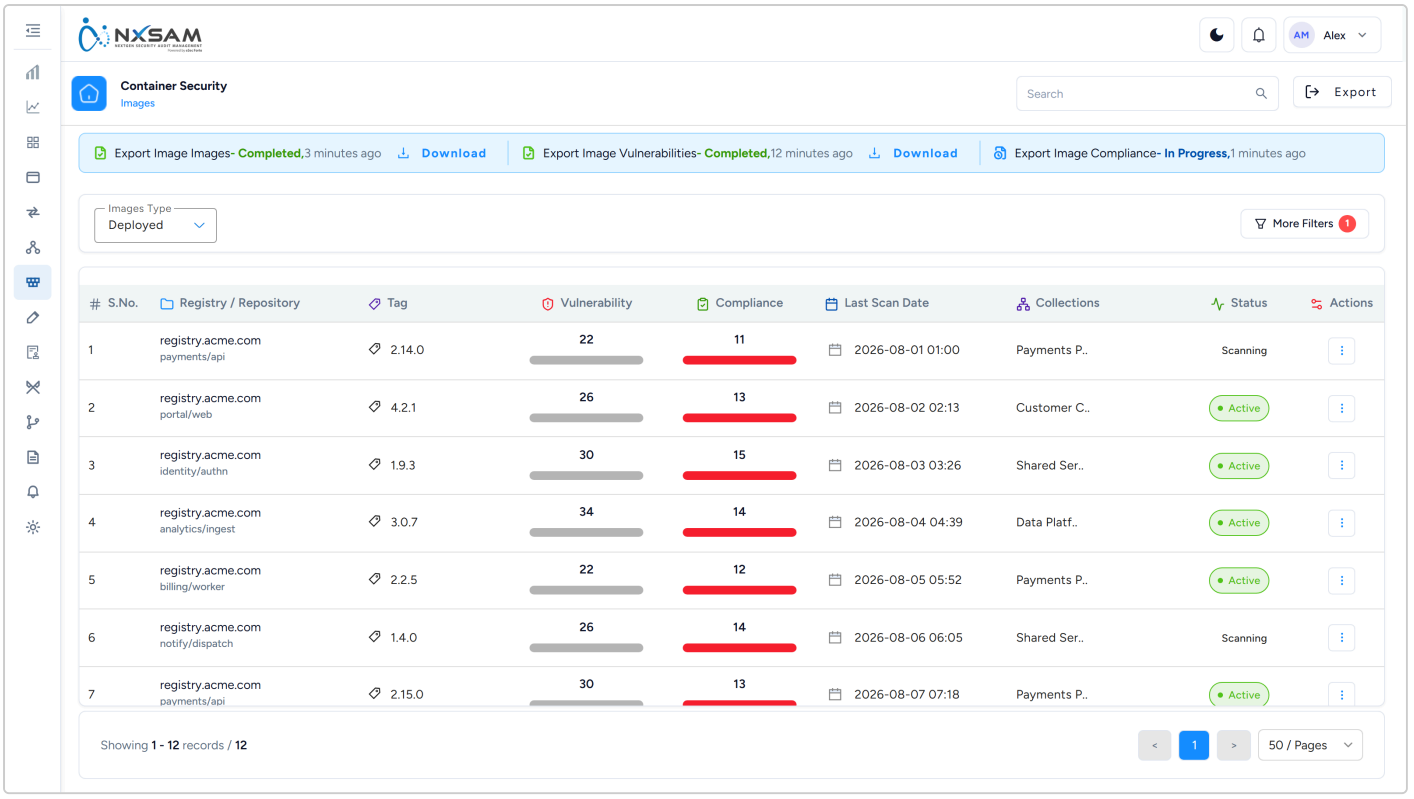


Container Security → Compliance

Container Images

The image inventory by repository and tag, with findings per image.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

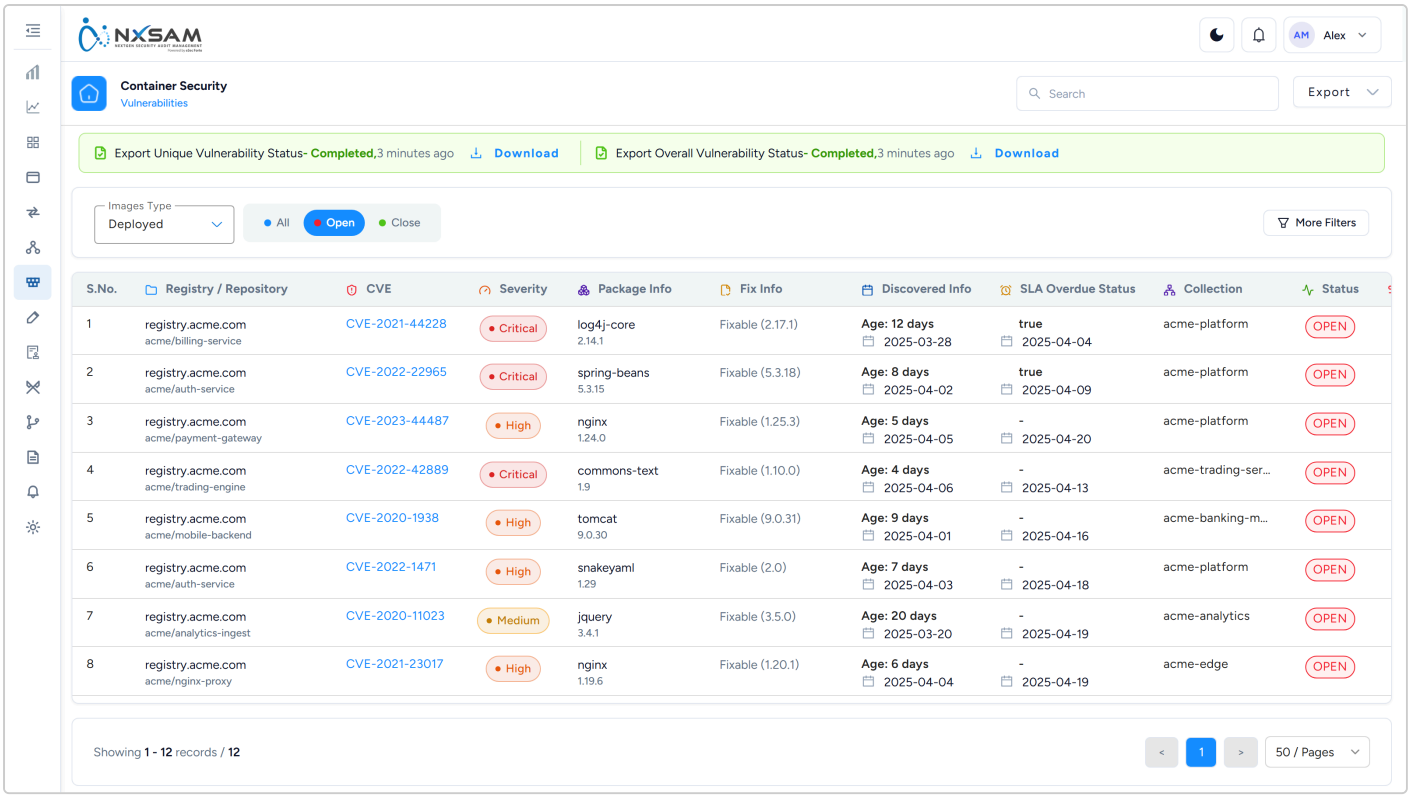


Container Security → Images

Container Vulnerability Search

Search findings across every image at once — the view that shows a shared base-image problem.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.



Container Security → Vulnerabilities

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Reference

Glossary

Terms as NxSAM uses them. Several are ordinary security words with a narrower meaning here, and those are the ones worth reading even if the term is familiar.

Assets and scope

Asset — anything that can hold a vulnerability: a web or mobile application, an API, a network host, a thick client, or a container image. Each type is a separate inventory with its own columns and its own licensed limit.

Asset property — an attribute recorded against an asset, such as criticality or whether it is internet-facing. Properties are more than description: they narrow SLA durations and feed the contextual risk score.

Bifurcation — splitting one application into separately assessed parts, so a large system can be tested module by module rather than as one unit.

Collection — a grouping of container images that comes from the scanning platform rather than from NxSAM, and is not editable here.

Scope — what an assessment covers. For applications this is the application; for compliance it is a host list, which is why compliance scope disputes are settled with an exported list of addresses.

Assessments

Audit — one assessment of one asset. It has a type, a category, a cycle, a scope, and a status of its own, separate from any finding's status.

Audit cycle — which round of a recurring programme an audit belongs to.

Confirmatory audit — a re-test created against a parent audit to verify that fixes hold. It is evidence about the original rather than an independent assessment, so the two should not be counted as two rounds of testing.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

Considered — the marker deciding whether an audit contributes to aggregate figures. Normally the latest audit per asset that reached a final state. An audit that is *not* considered still shows its findings on the module screen while contributing to no dashboard or report.

Initial audit — a first assessment, as opposed to a confirmatory one.

Findings

Vulnerability (also *finding*, *issue*) — one weakness on one asset, found by one assessment.

Vulnerability master — the catalogue entry (title, description, remediation) inherited by every occurrence of that vulnerability everywhere. Editing the master changes all of them.

Posture — Policy Compliance's unit: one control checked against one host. Every check produces a posture whether it passed or failed, which is why this module's volumes dwarf the others.

Artifact — the evidence attached to a finding: the steps, description and images that let someone else reproduce it.

Instance — which occurrence, where the same control or finding applies more than once to the same asset.

Status

Validation status — whether a finding has been confirmed. Nothing is counted anywhere until this reaches *Approved*, which is also when **Discovered On** is populated.

Client status (also *current status*) — what is being done about the finding. Of its values, only two close it.

Closed type — the flag deciding whether a status stops something counting as open. It is configuration rather than a fixed rule, so a deployment can change what "closed" means.

Posture status — what a compliance check returned: pass, fail or **error**. Error means the check could not run — a scanner or credentials problem rather than a compliance result.

Sub status — the qualifier beneath a main status: which kind of acceptance, or which stage of an approval.

Time and SLA

SLA duration — how long a finding of a given severity on a given asset type may remain open. Configurable, and narrowable by asset property.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

SLA expiry — the resulting deadline. Changing an SLA duration re-derives expiry dates, which can make a batch of findings overdue with no data change at all.

Discovered On — when a finding was first observed. Blank until the finding is approved.

Last observed — the most recent sighting, which is how a finding is known still to exist.

Overdue — past SLA expiry and not closed.

Decisions

Deviation (also *risk acceptance*, *RA*) — a recorded decision not to fix something, with an approval chain and an expiry. It does **not** close the finding: the exposure is still live and still counts as open.

Exception — a status marking an agreed exception rather than a resolution. Reporting that conflates the two overstates remediation.

Auto reject end date — when a temporary acceptance lapses on its own.

Integrations and processing

Integration — a connection to a scanning tool, either pulling from its API or parsing an uploaded report.

Vulnerability key — the columns deciding whether an incoming finding is one already seen. Changing it redefines "the same finding" and splits history.

Queue — asynchronous work: imports, syncs, report generation. A screen that seems not to have updated usually has a queue entry that explains it.

Data process — how far a tool-fed audit has got through the import pipeline; the difference between "the upload finished" and "the data is ready".

Beat — a scheduled job maintaining derived data, such as recalculating SLA states as dates pass. A stopped beat produces figures wrong by a fixed interval and raises no error.

Governance

RLCM — Red Line Compliance Management: assessment of controls against frameworks, rather than of systems for weaknesses.

TPRM — Third-Party Risk Management: the same control assessment applied to suppliers. It shares screens with RLCM and differs by the audit's type.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.

UCF — Unified Control Framework: the mapping that lets one control satisfy several frameworks at once, so answering it once covers each of them.

Control — a requirement being assessed, as opposed to a weakness being found.

Standard — the framework a control belongs to, such as ISO/IEC 27001 or PCI DSS.

Reporting

Considered audit filter — the switch on Report Search that reproduces either the dashboard's numbers or the module screens'. Comparing the two settles most disagreements about counts.

Design template — the layout a report renders through, separate from the template that decides its content.

Share view against share copy — sharing a template as a view keeps one definition that your edits reach; sharing a copy creates an independent template that will drift.

Copyright 2010-2026 eSec Forte® Technologies Private Ltd.