



HOUSEBLEND / RESEARCH & PRACTICAL GUIDANCE

NetSuite Integration Register: Owners, Credentials, Controls

Make NetSuite work better for your finance and operations teams with Houseblend. We help design, implement, integrate and improve ERP systems, with practical support for the people who use them every day.

Published by Houseblend · 2026-10-27 · netsuite integration register · netsuite integration inventory · netsuite integration governance · netsuite token based authentication · netsuite integration owner · netsuite decommission integration · netsuite restlet inventory · netsuite sandbox refresh · netsuite integration audit

Original article: <https://www.houseblend.io/articles/netsuite-integration-register-owners-credentials-decommission>



In this article

1. Executive Summary
 2. Introduction and Background
 3. Key Changes Affecting the Register
 4. What NetSuite Natively Inventories, and What It Does Not
 5. Register Design: Fields, Owners and the Orphan Test
 6. Credentials, Roles and Integration Records
 7. Sandbox, Schedules, Dependencies and Monitoring
 8. Implementation Considerations and Process Changes
 9. Data Analysis and Evidence
 10. Implications and Future Directions
 11. Frequently Asked Questions (FAQs)
 12. Conclusion
-

Executive Summary

A NetSuite integration register is the operational record that lets an organization identify, run, secure, transfer and retire every integration without depending on one person's memory. NetSuite's native tools do not supply that record on their own. Oracle's integration record "represents any app that sends web services requests or calls RESTlets" (docs.oracle.com), but "some integration record capabilities are limited for RESTlet activity" (docs.oracle.com). The RESTlet execution log "only shows data from those using token-based auth" (docs.oracle.com), and user credential (NLAAuth) calls "aren't logged on any integration record" (docs.oracle.com). Comma separated value (CSV) imports, scheduled scripts and file drops sit outside integration records, so a register must be assembled from several NetSuite lists plus the owners' own knowledge.

Three lifecycle facts raise the stakes as of October 2026. Oracle states that SOAP (Simple Object Access Protocol) web services will be removed with the 2028.2 release (docs.oracle.com), and that new SOAP integrations (docs.oracle.com) and new token-based authentication (TBA) integrations (docs.oracle.com) stop being possible from 2027.1. Existing TBA integrations are tentatively supported until 2028.2 (docs.oracle.com). A register that records channel and authentication method per integration turns those dates into a work list.

Credential and evidence facts drive most fields. TBA tokens "do not expire" (docs.oracle.com), and Oracle calls revocation after some time "a good practice" (docs.oracle.com), so credential age must be tracked by the organization. Resetting consumer credentials "invalidates the old ones" (docs.oracle.com), and a sandbox refresh deletes sandbox token keys and secrets (docs.oracle.com). Native activity logs are short: the SOAP execution log keeps 21 days in production (docs.oracle.com) and script logs 30 days, or 60 days for system error logs (docs.oracle.com), while system notes are not purged on a schedule (docs.oracle.com).

The recommended design is one row per integration, an accountable business owner and technical owner for each, credential metadata kept apart from secrets, a quarterly orphan test (an active integration with no reachable owner), and a decommission sequence with sign-off. The National Institute of Standards and Technology (NIST) SP 800-53 catalog supplies control language for inventories with owners

(raw.githubusercontent.com) and for least privilege (raw.githubusercontent.com). This report is documentation-based: product behavior, permissions and log coverage must be retested in a current account, and Table 2 includes fields to record that test.

Introduction and Background

An interface list names the connections between systems. A register answers different questions: who is accountable for each integration, why it exists, which records move in which direction, under which identity, on what schedule, with which recovery contact, and when it will be retired. The distinction matters most for mature accounts that have accumulated CSV jobs, scripts, bundles, middleware connections and vendor SuiteApps over many years. This report examines how to build and operate that register in NetSuite, using Oracle's own documentation for platform behavior and independent standards for control design.

The audience is the group that inherits or governs such an account: chief information officers, controllers, platform owners, security teams, internal audit and managed-support leaders, who must decide whether every integration can be identified, operated, secured, transferred and retired without one person's memory.

Scope is deliberate. API method details, OAuth (Open Authorization) setup steps and concurrency limits are covered in Oracle's own pages and in separate operating references, and are not repeated here. Concurrency is governed at account level for web services and RESTlets together (docs.oracle.com), so it is recorded as a register dependency rather than explained.

Oracle's overview lists four integration channels: CSV import, REST (Representational State Transfer) web services, RESTlets and SOAP web services, and says that "SuiteTalk REST web services with OAuth 2.0 authentication is the preferred technology for new integrations" (docs.oracle.com). Independent guidance points the same way on the need for an inventory. The OWASP (Open Worldwide Application Security Project) API Security Top 10 observes that a "lack of assets inventory and retirement strategies leads to running unpatched systems" (owasp.org), and CISA (the US Cybersecurity and Infrastructure Security Agency) calls asset visibility "a basic pre-condition" for managing cybersecurity risk (www.cisa.gov); that directive binds US federal agencies, so it is cited as rationale only.

Disclosure: this article is published by Houseblend, a Montréal NetSuite consultancy that is one possible provider of the stewardship described here; the same register can be built and run in house. Where professional opinion is needed, for example on records retention obligations, readers should consult internal audit, legal counsel or compliance.

Key Changes Affecting the Register

Four documented platform changes determine which fields a register must carry in 2026 and 2027. Dates below are Oracle's announced timeline as of October 2026 and may change; recheck them against current release notes before planning. Oracle also states that from the 2026.1 release all newly built integrations should use REST web services with OAuth 2.0, and that from 2027.2 only the last SOAP endpoint is supported, with older endpoints available but unsupported until SOAP removal in 2028.2 (docs.oracle.com).

SOAP web services are being retired

Oracle's SOAP removal FAQ states that "The 2025.2 SOAP endpoint is the last planned SOAP endpoint" ([docs.oracle.com](#)), that "only the last endpoint will be supported and have any potential bugs fixed" ([docs.oracle.com](#)), and that support continues "until the final removal of SOAP web services with the 2028.2 NetSuite release" ([docs.oracle.com](#)). At that point endpoints "will be disabled, and SOAP-based integrations will stop working" ([docs.oracle.com](#)). A register therefore needs a channel field and, for SOAP, the endpoint (WSDL, Web Services Description Language) version in use. Oracle's Application Performance Management (APM) dashboard can surface integrations that "still use unsupported WSDL versions" ([docs.oracle.com](#)), which gives the register a native cross-check.

New TBA integrations end in 2027.1

Oracle's TBA end-of-support page states that "you will no longer be able to create new integrations that use TBA" ([docs.oracle.com](#)). Support for existing integrations ends later, "tentatively NetSuite 2028.2" ([docs.oracle.com](#)). Oracle also notes that "third-party and app connectors that still rely on TBA may require an updated configuration" ([docs.oracle.com](#)). The register should hold authentication method as a first-class field so TBA integrations can be sorted by criticality and migration date.

RESTlets lose user credential support in 2027.1

For RESTlets, Oracle states that "support for user credentials for RESTlets ends in NetSuite 2027.1" ([docs.oracle.com](#)) and that "support for TBA for RESTlets ends in NetSuite 2028.2" ([docs.oracle.com](#)). A RESTlet caller using user credentials is both the least visible in native logs ("aren't logged on any integration record" ([docs.oracle.com](#))) and the first to reach end of support, so discovery relies on script deployment records and the callers' owners.

OAuth 2.0 is the direction for new integrations

Oracle's FAQ states that "only new integrations must use OAuth 2.0, starting with the 2027.1 NetSuite release" ([docs.oracle.com](#)), so existing REST integrations on TBA are not forced to migrate immediately. OAuth 2.0 is not available everywhere: "OAuth 2.0 cannot be used with SOAP web services" ([docs.oracle.com](#)). The register records which flow each integration uses, because credential artifacts differ.

What NetSuite Natively Inventories, and What It Does Not

The register cannot be generated from one report. Oracle's documentation describes several inventories, each covering a different slice, and some channels have no integration record at all. *Table 1* below summarizes the channel coverage found in Oracle's documentation as of October 2026.

Channel or artifact	Native inventory list	Native activity evidence and retention (Oracle)	Register implication
SOAP web services	Integration records, listed at Setup, Integration, Manage Integrations; viewing needs the Integration Applications permission (docs.oracle.com)	Per-record execution log: 21 days in production (docs.oracle.com), seven days in sandbox (docs.oracle.com)	Export record details on a schedule; the log cannot serve as an inventory
REST web services	Integration records (OAuth 2.0 or TBA)	Per-record REST log matching "requests that are uniquely identified with the integration record" (docs.oracle.com); Oracle's page states no retention period	Record consumer key owner and scope; confirm log behavior in a test account
RESTlets	Script deployment records ("When you deploy a script, NetSuite creates a script deployment record" (docs.oracle.com)) plus integration records for token callers	Integration record log "only shows data from those using token-based auth" (docs.oracle.com); user credential calls are not logged on any record (docs.oracle.com)	Treat script deployments as the primary inventory; document callers by interview
CSV import	Saved CSV Imports page: "shows all the saved import mappings you can access" (docs.oracle.com); no integration record	Job status kept "for 30 days" (docs.oracle.com)	Register each scheduled or programmatic job (docs.oracle.com) and its owner
Scheduled and map/reduce scripts, Suitelets	Script deployment records	Script Execution Log 30 days (docs.oracle.com); status pages delete details after 30 days (docs.oracle.com)	Capture the "execute as" identity and schedule per deployment; flag any Suitelet "available without login" (docs.oracle.com)
Bundles and SuiteApps	Installed Bundles page, which records "the user who installed the bundle" (docs.oracle.com)	Not applicable	Record hidden in-account dependencies and installer
SFTP (Secure File Transfer Protocol) file drops	No native inventory documented by Oracle	No native log documented by Oracle	Document from partner configuration and interviews

The native inventories are partial and mostly short lived. The register is the durable layer: its fields are refreshed from the native lists on a schedule and extended with facts no native list holds, such as business purpose, owner and recovery contact. RESTlet callers using user credentials, and any file drop, depend on human knowledge, which is the memory risk the register is meant to remove. Coverage claims in this table come from Oracle documentation and have not been retested; the test record in Table 2 is meant to close that gap.

Discovery sources, in order of reliability

- **Manage Integrations list:** the starting point for SOAP, REST and token-based RESTlet callers. Oracle notes that "each application can be represented by its own integration record" (docs.oracle.com), so one row per application is a reasonable register unit.

- **Script deployments and the Scripted Record page:** the latter lists records that have scripts "associated with a record" (docs.oracle.com); deployment records cover RESTlets, Suitelets and scheduled scripts.
- **Saved CSV Imports and job status:** anyone with Job Status page access "can see all import jobs" (docs.oracle.com).
- **Installed Bundles:** the list shows who installed each bundle (docs.oracle.com), and SuiteScript can query installed bundle information programmatically (docs.oracle.com). Middleware vendors document in-account dependencies, such as a SuiteApp Boomi installs by bundle ID (help.boomi.com) and one Zapier states "must be installed from the SuiteApp Marketplace" (help.zapier.com).
- **Middleware consoles:** connections are created outside NetSuite. Celigo documents a mode in which its platform generates the role and tokens (docs.celigo.com), and Make notes that "only admin can obtain the Consumer Key and Secret values and assign roles" (apps.make.com), so the middleware owner and the NetSuite owner are often different people.
- **Auto-installed records:** Oracle notes that "any user with account access can trigger auto-installation, not only account administrators" (docs.oracle.com), and that the Require Approval preference governs whether an installed record is enabled automatically (docs.oracle.com). The list therefore needs periodic review.

Register Design: Fields, Owners and the Orphan Test

Why one row per integration, and what a row contains

The register unit is one integration: a business purpose served through one channel under one identity. One vendor relationship can produce several rows (for example a nightly CSV import and a REST connection) because they fail, rotate and retire independently. *Table 2* below gives the recommended template. It combines fields Oracle's pages make available with fields only the organization can supply. NIST's CM-8 discussion lists "software owners, software version numbers, hardware inventory specifications, software license information" among the inventory attributes (raw.githubusercontent.com), and OWASP advises that teams "inventory all API hosts and document important aspects of each one" (owasp.org).

Field group	Fields	Source and control basis
Identity	Register ID; integration name; lifecycle state (proposed, active, frozen, retired); NetSuite application ID where an integration record exists; vendor or builder	Integration record fields; the application ID "isn't listed in system notes when the record is saved" (docs.oracle.com), so capture it at creation
Purpose and owners	Business purpose; accountable business owner; accountable technical owner; backup technical owner; recovery contact and escalation path	OWASP asks for a business justification and approval of each flow (owasp.org); one accountable person per item keeps authority clear (it.cornell.edu)
Channel and data	Channel (CSV, REST, RESTlet, SOAP, SFTP, middleware); direction; record types; volume and schedule; control totals used for reconciliation; downstream consumers	NIST AC-4 covers authorizations for "the flow of information within the system and between connected systems" (raw.githubusercontent.com)

Field group	Fields	Source and control basis
Identity and credentials	Integration user; role; authentication method; credential artifact types; credential created and last rotated dates; custodian of the secret store	A token pairs a user with an integration record (docs.mulesoft.com); no secret values are stored in the register
Environments	Production and sandbox account IDs; sandbox endpoint; connection per environment; post-refresh actions	Each account "has a unique account-specific domain" (docs.oracle.com)
Dependencies	Bundles or SuiteApps; scripts and deployments; saved searches; upstream and downstream jobs; middleware connection	Installed Bundles list (docs.oracle.com); script deployment records (docs.oracle.com)
Monitoring and recovery	Alert route; reconciliation job; log source and retention; restart procedure; last recovery test date	APM shows "the last 30 days" of data (docs.oracle.com)
Lifecycle and evidence	Review date; sunset trigger; decommission ticket; sign-off; evidence location and retention period	NIST AU-11 retention supports investigations and "regulatory and organizational information retention requirements" (raw.githubusercontent.com)
Test record	NetSuite account type (production, sandbox, Release Preview); enabled features; role; channel; test date; result	Product behavior and log coverage should be retested before the register relies on them

Table 2 deliberately excludes secrets. The register holds credential metadata (type, owner, dates), never token values; Oracle states that "credentials are not shown in system notes" ([docs.oracle.com](#)), and the register should not become a second place where they are exposed. The test record group matters because coverage facts in Oracle's pages depend on channel, authentication method and account type; recording account, features, role, channel and date lets a later reader reproduce the observation.

Business and technical owner model

Each integration needs two named humans, not a team alias. The accountable business owner, usually a controller or functional lead, decides whether the integration should exist and signs off its retirement. The accountable technical owner keeps it running, holds the recovery runbook and is the contact when credentials rotate. The RACI (Responsible, Accountable, Consulted, Informed) convention supplies the vocabulary: the University of Illinois toolkit describes Accountable as "takes ownership of the task" ([www.aitis.uillinois.edu](#)), and Cornell defines Responsible as the role performing "correct execution of the task/deliverable/decision" ([it.cornell.edu](#)). *Table 3* below applies it to register activities.

Activity	Business owner	Technical owner	NetSuite platform owner	Security and internal audit	Implementation or managed-support partner
Approve a new integration and its purpose	A	R	C	C	C
Create integration record, role and credentials	I	A / R	R	C	R (when contracted)
Maintain register row and review date	C	A / R	R	I	R (when contracted)
Rotate or revoke credentials	I	A / R	R	C	R (when contracted)

Activity	Business owner	Technical owner	NetSuite platform owner	Security and internal audit	Implementation or managed-support partner
Quarterly orphan test and attestation	A (for own rows)	R	R	C (audit evidence)	C
Decommission and sign-off	A	R	R	C	R (when contracted)

In *Table 3*, **A** marks the single accountable role per row. A partner column is shown because many organizations outsource part of this work, and the accountable role stays with the organization's own owners even when execution is contracted out. The table is a starting template to adapt to the organization's governance model.

The orphan test

An orphan is an active integration, credential or identity with no business owner or no technical owner who is currently employed and reachable. The test is a query, not a judgment: list every row whose lifecycle state is active and whose business or technical owner field is empty, names a departed person, or names an inactive NetSuite user. NIST's account management control lists accounts "no longer associated with a user or individual" as a trigger to disable them ([raw.githubusercontent.com](https://raw.githubusercontent.com/guides/owasp-asset-inventory/master/owasp-asset-inventory-1.0.0.pdf)), and OWASP's non-human identity (NHI) guidance warns of identities that "remain active without a new designated owner" ([owasp.org](https://owasp.org/asset-inventory/)) and recommends that teams "recertify NHIs to confirm they are still in use, actively needed, and have valid owners" ([owasp.org](https://owasp.org/asset-inventory/)), which is the quarterly attestation described below.

Credentials, Roles and Integration Records

Which artifacts exist, by channel

Credential artifacts differ by channel and must be recorded accordingly. For TBA, MuleSoft's documentation describes the Token ID as one "that represents the unique combination of a user and an integration record" (docs.mulesoft.com). For OAuth 2.0 client credentials, Workato documents that the flow "requires an EC private key and certificate" (docs.workato.com), and Oracle caps active certificates at five per combination of integration, entity and role, noting that an expired or revoked certificate means "you must create a new mapping for the integration record" (docs.oracle.com). For interactive OAuth 2.0 flows, users "can only list their own OAuth 2.0 authorized applications" (docs.oracle.com), while a holder of the management permission can "view or revoke any OAuth 2.0 authorized applications in the account" (docs.oracle.com). The register's credential fields therefore name the artifact (token, certificate, authorized application), the identity it binds to, and the custodian.

Credential age without invented thresholds

Oracle publishes no rotation interval for TBA tokens. It states that "TBA tokens do not expire" (docs.oracle.com) and that "it is a good practice to revoke a token after some time" (docs.oracle.com). Make's documentation likewise notes that "Make does not refresh the connection with NetSuite" (apps.make.com). OAuth 2.0 access tokens are short lived ("valid for 60 minutes" (docs.oracle.com)), so the long-lived artifacts are the client certificate, the consumer secret and any stored token.

Authoritative standards do not prescribe a universal interval either. NIST SP 800-53 leaves the period to the organization: its authenticator control requires that "a time period for changing or refreshing authenticators by authenticator type is defined" ([raw.githubusercontent.com](#)). NIST SP 800-63B addresses human passwords and says verifiers should not "require subscribers to change passwords periodically" ([pages.nist.gov](#)), on the stated basis that "humans have a limited ability to memorize complex, arbitrary secrets" ([pages.nist.gov](#)); that rationale does not transfer automatically to machine credentials. OWASP's non-human identity guidance flags secrets with "expiration dates that are too far in the future" ([owasp.org](#)). The register therefore carries three neutral fields: created date, last rotated date and an organization-defined rotation interval set by policy and risk rating, not copied from this report. Age is the review date minus the last rotated date, and a row with no recorded date is reported as unknown rather than compliant.

What rotation and reset actually do

Resetting an integration record's consumer credentials "invalidates the old ones" ([docs.oracle.com](#)), while "you can still use existing tokens and token secrets" with the new consumer pair ([docs.oracle.com](#)), so every holder of the old pair must be updated in the same change window. Lost token values cannot be recovered ("you will need to create a new token" ([docs.oracle.com](#))), and Boomi adds that "after leaving the confirmation page, you cannot retrieve these values" ([help.boomi.com](#)). Revocation is permanent ("inactive forever" ([docs.oracle.com](#))), although the token record stays "accessible for auditing purposes" ([docs.oracle.com](#)). Credentials cannot be reset on bundle-installed records ([docs.oracle.com](#)). Oracle states that "password rotation policies in your account don't apply to tokens" ([docs.oracle.com](#)), so token rotation is a separate, organization-run process. Its NetSuite Connector guidance warns "don't delete or inactivate the user ID to avoid breaking the connection" ([docs.oracle.com](#)); Oracle's documentation does not give a general statement on every effect of deactivating a token user, so the register records the token user and the retest result rather than assuming behavior.

Roles and least privilege

NIST AC-6 states the principle ("employ the principle of least privilege, allowing only authorized accesses for users" ([raw.githubusercontent.com](#))) and its discussion extends it to processes that "operate at privilege levels no higher than necessary to accomplish organizational missions or business functions" ([raw.githubusercontent.com](#)). For NetSuite, the register records the integration role per row. Oracle distinguishes token permissions: a user with Access Token Management "can create, assign, and manage a token for other users" ([docs.oracle.com](#)), and the User Access Tokens permission "does not also need the Log in using Access Tokens permission" ([docs.oracle.com](#)). Channel constraints apply too: "roles with the Web Services Only restriction will not work with RESTlets" ([docs.oracle.com](#)).

Scheduled scripts carry a different identity. For scheduled script deployments the execute-as role "is always set to Administrator and can't be changed" ([docs.oracle.com](#)), while other script types can run as "Current User, which runs the script with the logged-in user's permissions" ([docs.oracle.com](#)). A row for a scheduled script therefore records script owner and the privilege the script runs with, even though no integration user exists. OWASP warns of "lack of auditing and accountability due to indistinguishable activity between humans and automation" ([owasp.org](#)), which supports dedicated integration users, and Workato similarly states "we recommend creating a separate user for your integrations" ([docs.workato.com](#)).

Audit evidence for credential changes

A credential reset "is logged in the system notes as a Token-Based Authentication Credentials field update" (docs.oracle.com). Limits apply: system notes "are only created in the account where changes are made" (docs.oracle.com), so evidence for a shared record must be collected from each account. The Login Audit Trail can be used "to track TBA tokens and users" (docs.oracle.com), but Oracle's documentation states no retention period for it, so the register records where exported evidence is stored rather than assuming native retention.

Sandbox, Schedules, Dependencies and Monitoring

Sandbox and production differences

A register that lists production only will be wrong after the first refresh. Oracle's integration record guidance explains the shared ownership model: a record created in sandbox and installed in production means "you won't own it in production" (docs.oracle.com), and in the receiving account "you can only edit the State and the Note fields" (docs.oracle.com). Consumer credentials can be regenerated only in the originating account (docs.oracle.com), and Oracle warns that "regenerating Consumer Key or Consumer Secret in sandbox can break production integrations if shared" (docs.oracle.com). The register's environment fields therefore record where each integration record originated and whether it is shared.

Refresh effects are documented in several places. Production integration records "are copied to sandbox with their settings" (docs.oracle.com), and a record that existed only in sandbox "is removed during the refresh" (docs.oracle.com). TBA credentials do not survive: "after a sandbox refresh, Token Key and Token Secret are deleted and must be regenerated" (docs.oracle.com), and OAuth 2.0 users "must authorize applications explicitly in the sandbox" after each refresh (docs.oracle.com). For client credentials, "each time a sandbox account is refreshed, the setup gets cleared" (docs.oracle.com). Scheduled scripts are treated conservatively: "scripts scheduled in production are disabled by default in sandbox accounts" (docs.oracle.com), while scheduled map/reduce scripts "are copied to sandbox accounts, but are not scheduled by default" (docs.oracle.com).

Middleware vendors document the same pressure: Make notes that NetSuite "will regularly rollback sandbox environments to their original state" (apps.make.com), and Boomi advises that teams with several accounts or sandboxes "use a separate connection for each" (help.boomi.com). Each environment has its own endpoint, since each account "has a unique account-specific domain" (docs.oracle.com).

A post-refresh checklist for each integration row follows from these facts:

- **Review records:** block integration records that are unused or unwanted in sandbox, since copied records arrive with their settings.
- **Reissue credentials:** regenerate TBA tokens, repeat OAuth 2.0 authorizations and recreate client credential mappings, recording new created dates.
- **Re-enable schedules deliberately:** scheduled scripts arrive disabled and map/reduce deployments unscheduled; enable only what the test plan needs.

- **Check the shared-record rule:** do not regenerate consumer credentials in sandbox for a record shared with production.
- **Update the register:** record the refresh date, the person and the result.

Schedules and dependencies

The register's schedule fields capture when an integration runs and what it depends on. Native scheduling facts are scattered. Scheduled Script Status page details are deleted "after 30 days" (docs.oracle.com), and Map/Reduce status details are likewise deleted after 30 days (docs.oracle.com). CSV imports share queues, and "accounts that have purchased a SuiteCloud Plus license, have five queues for import jobs" (docs.oracle.com), so a register that records the queue and the order of dependent imports prevents silent overlaps.

Hidden dependencies deserve their own rows. Celigo documents that "as this bundle is a managed bundle, we will push the bundle in your account" (docs.celigo.com), so the version running can change without an internal change ticket; record the bundle, its installer (docs.oracle.com) and the vendor contact. Oracle states that Suitelets available without login "should not be used in integration use cases" (docs.oracle.com), so any found during discovery is a review item.

Monitoring and reconciliation links

Each row should name where success and failure are observed, and which control totals prove completeness. Oracle's APM SuiteApp can "monitor performance of user event scripts, workflows, RESTlets, scheduled scripts, and Suitelets" (docs.oracle.com), and its dashboard "highlights anomalies across record pages, scripts, saved searches, integrations, and processors" (docs.oracle.com). It raises alerts on "high error rates for integrations" (docs.oracle.com). APM data covers a limited window, since the tools "show data from the last 30 days" (docs.oracle.com). Script logs are purged after 30 days (user-generated) or 60 days (system error) (docs.oracle.com), so monitoring evidence needed for a quarter-end or audit review must be exported to the organization's own store. Reconciliation (record counts and amount totals compared between source and NetSuite) is documented in the row as a control total linked to the owning team's procedure.

Implementation Considerations and Process Changes

Standing up the register

Implementation follows the discovery sources in order: export the Manage Integrations list and script deployments as draft rows; add Saved CSV Imports, Installed Bundles and middleware connections; interview process owners for integrations no native list shows, such as file drops and RESTlet callers using user credentials; assign two owners per row and run the first orphan test; and fill the credential fields with metadata only. Unknowns left by the first pass are themselves findings.

Several process changes follow. New integrations require a register row before an integration record or role is created; credential resets go through a change ticket that updates the row; sandbox refreshes trigger the checklist above; and leaver processes query for rows whose owner or token user is the departing person,

aligning with NIST's account lifecycle of "create, enable, modify, disable, and remove accounts" (raw.githubusercontent.com) and its recognition that account types include "service" accounts (raw.githubusercontent.com).

Quarterly review

The review has five steps and a recorded attestation by each business owner for their rows:

1. **Refresh from native lists:** compare the register to the current Manage Integrations list, script deployments, Saved CSV Imports and Installed Bundles.
2. **Run the orphan test:** list active rows with a missing, departed or inactive owner and assign or retire each one.
3. **Compute credential age:** report rows by days since last rotation against the organization's own interval, with unknown dates reported separately.
4. **Check lifecycle dates:** list SOAP rows, TBA rows and RESTlet user credential rows against the announced Oracle timeline, flagging any without a migration plan or sunset date.
5. **Review privilege and activity:** compare each integration role to its documented need, check that tokens still map to active users, and review system notes and audit trails, which Oracle recommends doing "regularly" (docs.oracle.com).

Decommissioning an integration

Decommissioning is a sequence, not an event. A suggested checklist, to be adapted to the organization's change process:

1. **Confirm intent:** the business owner approves retirement and the technical owner confirms dependents (downstream reports, bundles, scripts, other integrations).
2. **Freeze:** set the lifecycle state to frozen, announce a date, and stop schedules. For a SOAP caller, Oracle documents that an integration record can be used to "block SOAP web services requests" (docs.oracle.com), which gives a reversible step before deletion.
3. **Observe:** monitor for unexpected callers during the agreed window, using available logs while they last (21 days for the SOAP execution log in production (docs.oracle.com)).
4. **Revoke credentials:** revoke tokens, authorized applications and certificates; Oracle states revocation of a token is permanent, and revoking an OAuth 2.0 authorized application "invalidates all tokens associated with that application" (docs.oracle.com).
5. **Remove artifacts:** disable or delete script deployments, saved imports, roles, integration users, the middleware connection and any bundle that served only this integration.
6. **Handle data and secrets:** dispose of exported files and stored secrets according to policy; NIST's media sanitization control requires sanitization "prior to disposal, release out of organizational control, or release for reuse" (raw.githubusercontent.com).
7. **Check sandbox echoes:** production records are copied to sandbox (docs.oracle.com), and a refresh is one way sandbox-only records are removed (docs.oracle.com).
8. **Record evidence and sign-off:** store the change ticket, system notes export, revocation confirmations and owner sign-offs, then set the row to retired and keep it.

A retired row is retained, not deleted. System notes for existing records are "never automatically deleted" ([docs.oracle.com](#)), and revoked tokens remain "accessible for auditing purposes" ([docs.oracle.com](#)), so the native trail persists even though logs do not. The retention period for the register's own evidence is set by policy and applicable law; NIST AU-11 frames retention as supporting "after-the-fact investigations of incidents" ([raw.githubusercontent.com](#)). The reader should confirm required periods with legal counsel or internal audit rather than inferring them from this report.

Data Analysis and Evidence

The evidence in this report is documentary: platform facts that bound what a register can rely on. This report makes no prevalence claim because it draws only on vendor and standards documentation.

Retention windows (Oracle documentation, as of October 2026).

- **SOAP execution log:** 21 days in production ([docs.oracle.com](#)) and seven days in sandbox ([docs.oracle.com](#)); the SOAP usage log can purge "earlier than 21 days" under heavy activity ([docs.oracle.com](#)).
- **Script logs:** 30 days for user-generated logs and 60 days for system error logs ([docs.oracle.com](#)), capped at 5 million logs per database instance ([docs.oracle.com](#)). Oracle's same page also states that all script execution logs are stored for up to 30 days, so treat 30 days as the planning figure, confirm actual retention in a current account, and note that the volume cap can purge logs sooner.
- **CSV import status:** 30 days, up to 1,000,000 jobs ([docs.oracle.com](#)).
- **APM:** data from the last 30 days ([docs.oracle.com](#)).
- **System notes:** no scheduled purge for existing records ([docs.oracle.com](#)).

Credential lifetimes. TBA tokens do not expire ([docs.oracle.com](#)); OAuth 2.0 access tokens are valid for 60 minutes ([docs.oracle.com](#)); the refresh token validity setting for the code grant flow defaults to 48 hours ([docs.oracle.com](#)); client credential certificates are capped at five active per combination ([docs.oracle.com](#)). TBA is therefore the group where manually tracked credential age matters most.

Coverage and dates. Of the seven rows in Table 1, one (SFTP file drops) has no native inventory documented by Oracle and one (RESTlets) has a documented log limitation; this counts documentation findings, not a share of any account's integrations. The platform dates above convert into a count of affected rows once channel and authentication method are recorded. Two limits apply: Oracle's REST log page states no retention period, and the Oracle pages cited here show no revision date, so the as-of date for this article is 27 October 2026.

Implications and Future Directions

The immediate implication is a time-boxed migration inventory: a register with channel and authentication fields turns the announced 2027.1 and 2028.2 dates into counts and owners. Existing TBA integrations can continue under Oracle's stated timeline, so ordering is a business decision driven by criticality and vendor readiness, and Oracle's note that connectors relying on TBA may need updated configuration ([docs.oracle.com](#)) argues for including middleware vendors in the review.

The second is shared responsibility: Celigo states that inadequate role permissions "could lead to unwanted errors when executing the data flows" (docs.celigo.com), and Zapier describes a method in which "users only need their Account ID to connect" (help.zapier.com). Credential and ownership artifacts live in more than one system, so a NetSuite-only list will miss them.

The third is that evidence should be designed rather than discovered: because native logs are short, export system notes, integration record snapshots and monitoring summaries on a schedule, with retention set by policy. Oracle's timelines may shift, and log coverage differs by channel and authentication method. Organizations that prefer outside help can engage an implementation or managed-support partner for discovery and ongoing stewardship. The template here is usable by any team, whether run in house or with outside help.

Frequently Asked Questions (FAQs)

What is a NetSuite integration register, and how does it differ from an inventory template?

A register is the owned, reviewed version of an inventory: it adds accountable owners, credential metadata, lifecycle state, review dates and evidence to the list (Table 2).

How should integration owners be assigned?

Assign one accountable business owner and one accountable technical owner per row, with a named backup, and keep a single accountable role per activity (it.cornell.edu). Run the orphan test to catch rows whose owners left.

How do NetSuite TBA credentials get rotated?

Oracle publishes no interval: tokens "do not expire" (docs.oracle.com). Create a new token for the integration user and role, update the secret store, confirm traffic, then revoke the old token. Resetting consumer credentials "invalidates the old ones" (docs.oracle.com), so coordinate that change with every holder, record the dates and set the interval by policy.

How is data flow mapped for each integration?

Record direction, record types, schedule, control totals and downstream consumers per row; NIST AC-4 frames the control basis as information flow "between connected systems" (raw.githubusercontent.com). Use one diagram key (symbols for channel, direction and trust boundary) across the portfolio.

How do you decommission a NetSuite integration?

Follow the eight-step checklist above, from confirming intent through freezing, revoking credentials and removing artifacts to storing evidence with sign-offs.

What belongs on an integration audit checklist?

Every active row has two owners, a purpose, a channel and an authentication method; credential dates are recorded; roles follow least privilege ([raw.githubusercontent.com](#)); sandbox post-refresh actions were done; and system notes were reviewed ([docs.oracle.com](#)).

How should least privilege be applied to integration users?

Use a dedicated user and role per integration, grant only what the channel needs, and avoid permissions such as Access Token Management that let a user manage tokens for others ([docs.oracle.com](#)).

How are RESTlets and Suitelets inventoried?

Start from script deployment records ([docs.oracle.com](#)), then add callers by interview, because user credential calls "aren't logged on any integration record" ([docs.oracle.com](#)).

What is integration governance in this context?

The rules that keep the register accurate: a required row before go-live, quarterly owner attestation, credential and lifecycle fields, and decommission sign-off.

Conclusion

A NetSuite integration register is feasible, but it must be assembled and maintained deliberately because the platform's native inventories are partial and its native logs are short. Integration records, script deployments, Saved CSV Imports and Installed Bundles each cover a slice, and file drops and some RESTlet callers fall outside all of them. The register is the durable layer that joins these lists to owners, purpose, credentials and lifecycle.

The design choices that matter most are small and testable: one row per integration, two named owners, credential metadata without secrets, a computed credential age against an organization-defined interval, a sandbox checklist that reflects Oracle's documented refresh effects, and a quarterly orphan test with owner attestation. The announced changes to SOAP, TBA and RESTlet authentication give the register an immediate, dated purpose. As of October 2026, the remaining uncertainty is coverage by channel and account type, which the test record fields in Table 2 are meant to resolve. Each organization should retest the behaviors cited here in a current account and set retention and rotation intervals through its own policy and professional advice.

Source links

Links cited in this article, in order of first appearance. Full addresses are provided for readers using extracted text.

1. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/chapter_4399047360.html
2. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4389727047.html
3. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4639542510.html
4. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/article_2104046421.html

5. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_0525020842.html
6. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/bridgehead_4254081947.html
7. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4395597766.html
8. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1515141510.html
9. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4543338726.html
10. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N3352137.html
11. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N673560.html
12. https://raw.githubusercontent.com/usnistgov/oscal-content/main/nist.gov/SP800-53/rev5/json/NIST_SP-800-53_rev5_catalog.json
13. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1500275531.html
14. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1529089601.html
15. <https://owasp.org/API-Security/editions/2023/en/Oxa9-improper-inventory-management>
16. <https://www.cisa.gov/binding-operational-directive-23-01>
17. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_160274514817.html
18. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N2971402.html
19. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/chapter_4247329078.html
20. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_156570320384.html
21. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4486246754.html
22. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N353796.html
23. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N350581.html
24. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4659353312.html
25. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/chapter_N2997713.html
26. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N3397033.html
27. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/chapter_1494642209.html
28. https://help.boomi.com/docs/Atomsphere/Integration/Connectors/r-atm-NetSuite_connection_f6253acc-b896-421e-abfd-aebe741950f5
29. <https://help.zapier.com/hc/en-us/articles/42853962942989>
30. <https://docs.celigo.com/hc/en-us/articles/4413014718747-FAQ-Auto-generated-token-based-authentication-for-NetSuite-connection>
31. <https://apps.make.com/netsuite.md>
32. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4429293934.html
33. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4543308762.html
34. <https://it.cornell.edu/index.php/it-service-management/raci-and-rasci-definitions>
35. <https://docs.mulesoft.com/connectors/netsuite/netsuite-studio-configure>
36. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1498251763.html
37. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4283525918.html
38. https://www.aitis.uillinois.edu/services/professional_services/pmo/raci_toolkit
39. <https://owasp.org/www-project-non-human-identities-top-10/2025/1-improper-offboarding>

40. <https://docs.workato.com/connectors/netsuite-rest/connect.html>
41. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_162686838198.html
42. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_157960925955.html
43. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_157771510070.html
44. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_162755359851.html
45. <https://pages.nist.gov/800-63-4/sp800-63b.html>
46. <https://owasp.org/www-project-non-human-identities-top-10/2025/7-long-lived-secrets>
47. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/bridgehead_4249078810.html
48. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_162606561992.html
49. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/bridgehead_4248124361.html
50. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4254801119.html
51. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1508948655.html
52. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/chapter_N2997500.html
53. <https://owasp.org/www-project-non-human-identities-top-10/2025/top-10-2025>
54. <https://docs.workato.com/connectors/netsuite/netsuite-connect.html>
55. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1534941538.html
56. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1515139626.html
57. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_1516026974.html
58. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/subsect_164330096846.html
59. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_4478613244.html
60. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N347015.html
61. <https://docs.celigo.com/hc/en-us/articles/360015950192-Before-you-install-integrator-io-20038-bundle-in-NetSuite>
62. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/chapter_4283522055.html
63. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/chapter_158644279544.html
64. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_N3444819.html
65. https://docs.oracle.com/en/cloud/saas/netsuite/ns-online-help/section_157771733782.html
66. <https://docs.celigo.com/hc/en-us/articles/115000368671-NetSuite-connection-prerequisites>

THE PUBLISHER

About Houseblend

Houseblend is a NetSuite consulting firm serving finance and operations teams. We help organizations implement ERP systems, connect business applications, improve existing configurations and maintain the systems that support everyday work. Our audience includes finance leaders, controllers, operations managers, NetSuite administrators and implementation teams.

Implementation and architecture

Houseblend provides NetSuite implementation, architecture and data migration services. We help teams evaluate how business processes, reporting requirements and existing data should fit together in an ERP environment. Training supports the people responsible for adopting and operating the resulting system.

Integrations, customization and AI

Our services include NetSuite integrations and customization, as well as AI integrations and AI transformation work. These engagements connect ERP data and workflows with the broader application landscape. The right design depends on the organization's systems, controls and operating needs.

Improve and support an existing system

Houseblend offers NetSuite health checks, optimization, managed support and project rescue services. We also provide expertise for analytics and specialist workflows, including NetSuite Analytics Warehouse, warehouse management and field service management. Published educational material helps teams investigate options and prepare informed questions for their implementation or support work.

Work with Houseblend

Explore [NetSuite implementation](#), [integrations](#), [managed support](#) and [AI integrations](#). [Contact Houseblend](#) to discuss your current system and priorities.

Article examples explain concepts rather than promising a particular license, product capability, delivery schedule or outcome. Engagement scope is confirmed with the Houseblend team.

Website: <https://www.houseblend.io>

This article is educational. Verify consequential medical, legal, financial, regulatory and technical decisions with appropriate professionals and the cited primary sources. Company services are described separately from the article's research findings.