SNAKEZ S.R.O

Security and operational visibility

Spyderz Sense

User Guide

AWS deployment, scanning, evidence, and operations

For administrators, scanner operators, and read-only reviewers

FOR AUTHORISED ENVIRONMENTS ONLY

Published July 2026 | Snakez S.R.O | snakez.sk

Contents

Use this guide to deploy, configure, operate, and troubleshoot the current Spyderz AMI and web console.

This guide describes the current Spyderz AMI and web console behavior. It is intended for administrators, scanner operators, and read-only reviewers.

Product Scope

Spyderz is a local-first AWS EC2 inventory and vulnerability scanner. It:

- Discovers EC2 instances and attached security groups across enabled AWS regions.
- Checks whether instances are registered and online in AWS Systems Manager (SSM).
- Collects Linux and Windows inventory by sending non-interactive SSM commands.
- Matches installed operating systems and software against CVE data.
- Performs vendor-specific patch checks where supported.
- Correlates listening ports with public security-group ingress.
- Stores inventory, scan runs, findings, settings, users, and notifications in local PostgreSQL.
- Exports completed scan data as Excel workbooks.

Spyderz does not exploit targets. It does not perform credential attacks, malware execution, full file-system scanning, or full compliance benchmarking.

> Important: a zero-CVE result means no vulnerability survived the current scanner's collection, matching, threshold, and patch-validation logic. It is not proof that the host has no vulnerabilities.

License Tiers

The tier is compiled into the AMI and displayed beside `Spyderz Sense Console`. It cannot be changed from the UI.

Tier	EC2 target slots	Lifetime normal releases	Tag override
Standard	10	3	No
Advanced	100	10	No
Enterprise	Unlimited	Unlimited	Yes

Standard and Advanced use an SSM Parameter Store registry:

```text
/scanner/license/registry
```

Enterprise does not use target-slot enforcement.

AWS Deployment Requirements

Scanner AMI

Launch the selected Standard, Advanced, or Enterprise AMI with:

- Root volume: the AMI defaults to `gp3`, `33 GiB`.
- IMDS enabled. Licensing and region detection use IMDSv2.
- An EC2 instance profile containing the scanner permissions in this guide.
- Network access to AWS APIs and vulnerability data sources over HTTPS.
- Enough CPU and memory for the target count. Watch the live CPU and MEM indicators. Sustained yellow or red utilization means the scanner instance needs more capacity or a smaller scan batch.

Scanner security group

Recommended inbound rules:

Port	Protocol	Source	Purpose
443	TCP	Administrator/VPN CIDR only	Spyderz web console
22	TCP	Administrator/VPN CIDR only	Optional SSH administration

Do not expose PostgreSQL port `5432`. PostgreSQL is intended to remain on localhost.

Port `80` only redirects to HTTPS. It does not need to be publicly allowed when users connect directly to `https://`.

Outbound access

Allow outbound HTTPS to:

- AWS EC2, SSM, and optional RDS API endpoints.
- NVD CVE API.
- Microsoft Security Update Guide API.
- Ubuntu, Debian, and Red Hat security sources.
- CISA KEV, GitHub, and Exploit-DB enrichment sources.

Private subnets need NAT or suitable VPC endpoints. SSM-managed targets also need connectivity to the regional SSM service.

IAM Configuration

Attach a role to the Spyderz scanner instance. This functional baseline matches the AWS API calls made by the current code:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DiscoverAwsInventory",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstances",
        "ec2:DescribeRegions",
        "ec2:DescribeSecurityGroups"
      ],
      "Resource": "*"
    },
    {
      "Sid": "InspectAndScanSsmManagedNodes",
      "Effect": "Allow",
      "Action": [
        "ssm:DescribeInstanceInformation",
        "ssm:SendCommand",
        "ssm:GetCommandInvocation"
      ],
      "Resource": "*"
    },
    {
      "Sid": "ReadAndUpdateLicenseRegistry",
      "Effect": "Allow",
      "Action": [
        "ssm:GetParameter",
        "ssm:PutParameter"
      ],
      "Resource": "arn:aws:ssm:*:*:parameter/scanner/*"
    },
    {
      "Sid": "OptionalRdsSecurityGroupCorrelation",
      "Effect": "Allow",
      "Action": [
        "rds:DescribeDBInstances"
      ],
      "Resource": "*"
```

```
    }
  ]
}
```

Notes:

- `ssm:PutParameter` is required for Standard and Advanced license slot registration.
- `rds:DescribeDBInstances` is optional and is used only for best-effort security-group attachment/orphan correlation.
- A customer-managed KMS key protecting the license parameter also requires `kms:Decrypt` and `kms:Encrypt` permissions for that key.
- `ssm:SendCommand` is broad in this baseline. After confirming scans work, restrict it to approved target resources and these AWS documents: `AWS-RunShellScript` and `AWS-RunPowerShellScript`.
- AWS Organizations SCPs, permission boundaries, and explicit denies can still block these calls even when this role policy allows them.

Target Instance Requirements

Every target must meet all applicable requirements:

1. EC2 state is `running`.
2. SSM Agent is installed and running.
3. Target is registered as an SSM managed node.
4. SSM reports `PingStatus=Online`.
5. Target instance role includes `AmazonSSMManagedInstanceCore` or equivalent SSM permissions.
6. Target can reach regional SSM endpoints.
7. Standard/Advanced targets have the required tag and a registered license slot.
8. Linux targets have `python3`, `ss`, and normal package-manager commands.
9. Windows targets support PowerShell and the required CIM/network cmdlets.

No inbound SSH, WinRM, or RDP access from the scanner is required. Collection is performed through SSM Run Command.

Launch and First Login

SSH access

Use the launch key pair and the AMI user:

BASH

```
ssh -i /path/to/key.pem SpyderzUser@SCANNER_PUBLIC_IP
```

`SpyderzUser` has passwordless `sudo` in the AMI.

Web access

Open:

TEXT

```
https://SCANNER_PUBLIC_IP/
```

The AMI initially uses a self-signed TLS certificate. A browser warning is expected until a trusted certificate is installed.

Initial credentials

Fresh AMI credentials are case-sensitive:

TEXT

```
Username: Admin
Password: Admin
```

First login redirects to the Welcome screen. Complete all three steps:

1. Accept the Spyderz EULA checkbox.
2. Select timezone and 12-hour or 24-hour clock.
3. Enter and confirm a new password.

The onboarding flag is stored in PostgreSQL. Later logins use the new password and should open the dashboard without returning to Welcome.

If a source/manual installation shows `Spyderz setup` instead, create its first admin account and scanner defaults there.

Tagging and Scan Eligibility

Standard and Advanced enforce this case-sensitive EC2 tag:

TEXT

```
Key: Spyderz
Value: Sense
```

AWS CLI example:

BASH

```
aws ec2 create-tags \
  --region us-east-1 \
  --resources i-0123456789abcdef0 \
  --tags Key=Spyderz,Value=Sense
```

An asset is selectable only when it is:

- Running.
- Tagged correctly, unless Enterprise tag override is enabled.
- Online in SSM.
- Allowed by the active license.

After adding or removing tags, run `Refresh inventory`.

Enterprise tag override

Enterprise administrators can open:

TEXT

```
Settings > Effects > Override Tagging Policy
```

- `OFF`: `Spyderz=Sense` is required and the button says `Scan Tagged`.
- `ON`: tags are ignored for eligible running instances and the button says `Scan All`.

The default is `OFF`.

License Slot Behavior

Standard and Advanced slots track target EC2 instance IDs, not concurrent scan threads.

- A newly tagged target is registered when a slot is available.
- A registered and still-tagged target can be scanned again without consuming a new slot.
- Removing `Spyderz=Sense` skips the target but does not release its slot.
- Stopping or terminating a target does not automatically release its slot.
- When capacity is full, already registered and currently tagged targets remain eligible. Newly tagged, unregistered targets are skipped.
- The live preparation message can show, for example, `Preparing 10/15 targets for scanning, limited by license.`

Normal release allowances are lifetime counters:

- Standard: 3 releases.
- Advanced: 10 releases.

Slot release and one-time support override operations are administrative license actions. They are not currently exposed in the web UI. Contact Spyderz support or use the tier-specific supplied license utility. Do not edit the SSM JSON manually; its HMAC protects integrity.

If the UI reports:

```text
License registry not found in SSM. Please run setup_license.py or contact support.
```

initialize the registry with the tier-specific setup utility or contact support. Do not use `--force` on an existing registry unless instructed; it overwrites registrations and release history.

> Current limitation: Standard and Advanced target-slot resolution uses the scanner's effective AWS region. Keep licensed targets in that region unless the deployment has been validated for multi-region license registration.

Recommended First Run

1. Confirm the scanner role and target SSM roles.
2. Log in and finish onboarding.
3. Open `Settings > NVD key` and add an NVD API key.
4. Confirm `Settings > Defaults`.
5. Tag intended targets with `Spyderz=Sense`.
6. Open `Assets`.
7. Click `Refresh inventory`.
8. Review each region and resolve `No Access` or SSM Offline states.
9. Select one test Linux or Windows target.
10. Click `Scan Selected`.
11. Watch the Scans live transcript through completion.
12. Review Vulnerabilities, Security Groups, Dashboard, and Reports.
13. Expand to the remaining licensed fleet.

Starting with one target makes IAM, SSM, network, and package-collection errors easier to isolate.

Dashboard

Dashboard presents current database-backed posture:

- Fleet and scannable asset counts.
- License limit text for Standard and Advanced when tags exceed capacity.
- Scan activity trend.
- Severity split.
- Vulnerability pressure heat map.
- Top impacted packages.
- Recent vulnerabilities.
- Live posture and recent scan risk.

Dashboard panels need persisted inventory and completed scans. Empty charts before the first completed scan are expected.

Dashboard is an overview, not the evidence source. Use Vulnerabilities and Reports for asset/package-level evidence.

Assets and Inventory Refresh

Assets displays discovered Linux and Windows EC2 instances with:

- Name, instance ID, region, state, private IP, and detected platform.
- Tag and scan eligibility.
- SSM registration/readiness.
- Latest score and scan time.
- Critical count, listening-port count, user counts, groups, and status.

Use `All`, `Linux`, and `Windows` to change the visible platform set.

Refresh inventory

`Refresh inventory` is discovery and baseline inventory collection. It is not the same as a vulnerability scan.

It:

1. Enumerates enabled AWS regions, minus configured exclusions.
2. Discovers EC2 instances and attached security groups.
3. Checks SSM readiness.
4. Reuses or collects inventory snapshots where applicable.
5. Writes current assets, security-group attachments, and inventory to PostgreSQL.

The Assets page keeps a separate collapsible `inventory refresh log`. Region cards show:

- `OK`: instances were found and region calls worked.
- `SG - only`: security groups exist but no EC2 instances were found.
- `Empty`: no instances were found.
- `No Access`: AWS API access failed.

Region details include discovered instances, scannable count, security groups, reused snapshots, collected snapshots, skipped items, and errors.

Running a Vulnerability Scan

Scan Tagged

Scans all currently eligible `Spyderz=Sense` targets, constrained by license and SSM state.

Scan All

Enterprise only. Appears when `Override Tagging Policy` is enabled and scans all eligible running targets without requiring tags.

Scan Selected

Scans checked rows. Non-scannable rows cannot be selected.

After a scan starts, the UI waits approximately one second, redirects to Scans, and highlights the Scans navigation item.

Scan phases

Typical Linux flow:

1. Detect operating system/distribution.
2. Collect users, groups, ports, and running services.
3. Collect `dpkg` or `rpm` packages.
4. Match each package against CVE data.
5. Run vendor patch checks and enrichment.
6. Persist target results.
7. Finalize the stored run and report metadata.

Typical Windows flow:

1. Collect OS and security status.
2. Collect installed KB/hotfix data.
3. Collect 64-bit and 32-bit registry software.
4. Collect running services.
5. Collect TCP/UDP listeners and process mapping.
6. Collect local users and administrators.
7. Normalize inventory.
8. Match Windows OS and software against CVEs.
9. Check Microsoft patch status for accepted matches.
10. Persist target results and findings.

Scan Progress and Controls

Live transcript

The Scans page shows a collapsible terminal-style transcript. It remains open after completion for the current browser session. Live events include:

- Target selection and license limiting.
- SSM connection and collection sections.
- Package number, total, version, and match state.
- Patch checks.
- Database writes.
- Completion, warning, skip, stop, and failure messages.

Live transcript data is in memory. Restarting the API clears it. Stored scan runs remain in PostgreSQL.

Progress

- `Target %` is progress for one target.
- `Total progress` weights every target equally.
- With 10 targets, each completed target contributes 10% of total.
- Partial progress inside the current target contributes a fraction of that target's share.

Top live card

During a scan, the top bar displays:

- Current target.
- Current phase.
- Total progress.
- Elapsed minutes.
- Scanner-host CPU and memory percentage.

Resource indicator colors:

- Green: below 65%.
- Yellow: 65% through 84.9%.
- Red: 85% or higher.

Jump To Next Asset

Confirms that the current target should be skipped and advances to the next queued target. Data in the skipped action scope may be incomplete.

Stop Scan

Stop is cooperative:

- Spyderz marks the run for stopping.
- Pending targets stop being dispatched.
- Code checks the stop flag between collection and matching steps.
- An SSM command already sent to a target may continue until it returns or reaches its timeout.
- The run is stopped/aborted and is not treated as a completed reportable run.

Repeatedly clicking Stop does not make an already-dispatched SSM command return faster.

Stored runs

Expand a stored run to see:

- Scanned assets.
- Package, user, port, and service counts.
- CVE and critical counts.
- Peak risk.
- Stored report and inventory output paths.

Collected Data

Linux

Supported package inventories:

- Ubuntu and Debian: `dpkg-query`.
- Amazon Linux, RHEL, CentOS, Fedora, Rocky, AlmaLinux, Oracle Linux, SUSE, and openSUSE: `rpm`.

Linux inventory includes:

- Hostname.
- Installed package names, versions, and package manager.
- Local users, UID, shell, and login/system/service classification.
- Local groups, GID, and explicit members.
- TCP/UDP listening addresses from `ss -tuln`.
- Running systemd service unit names.

Windows

Windows data is collected in independent sections. A missing cmdlet records a collection error instead of intentionally failing unrelated sections.

Operating system

- Computer name.
- Windows product/caption and edition.
- Version and build number.
- Architecture.
- Install date.
- Last boot time.

Sources: `Get-ComputerInfo`, `Win32_OperatingSystem`, and `Win32_ComputerSystem`.

Updates and hotfixes

- KB/HotFix ID.
- Description.

- Installed date.
- Installed by.

Source: `Get-HotFix`.

Installed software

- Display name and version.
- Publisher.
- Install date and location.
- Uninstall string.
- 32-bit or 64-bit registry source.

Sources:

```text
HKLM:\Software\Microsoft\Windows\CurrentVersion\Uninstall\*
HKLM:\Software\WOW6432Node\Microsoft\Windows\CurrentVersion\Uninstall\*
```

Spyderz does not use `Win32_Product`, avoiding MSI repair/reconfiguration side effects.

Services

For running services:

- Name and display name.
- State and start mode.
- Run-as identity.
- Executable command/path.

Current service posture checks include:

- Unquoted executable paths containing spaces.
- Services running as LocalSystem/SYSTEM.

Listening ports

- TCP listening endpoints.
- UDP endpoints up to port 49151.
- Local address and port.
- State.
- PID.
- Process name and path when accessible.

Windows posture findings call out common ports including RDP `3389`, SMB `445`, WinRM `5985/5986`, SQL Server `1433`, and web `80/443`.

Users and administrators

- Local users.
- Enabled state.
- Last logon.
- Password last-set time.
- Members of the local Administrators group.
- Object class and principal source where available.

Defender and firewall

- Defender enabled state.
- Real-time protection state.
- Antivirus signature version and update time.
- Domain, Private, and Public firewall profile state.

Windows findings versus CVEs

Windows scan messages report two different totals:

- `Windows security finding(s)`: posture issues such as open management ports, disabled protection, local-admin count, or risky services.
- `CVE vulnerability match(es)`: accepted software/OS CVEs after matching and patch checks.

Example: `14 Windows security findings and 2 CVE vulnerability matches` does not mean all 14 findings are CVEs.

Vulnerability Matching

The scanner keeps generic CVE matching separate from vendor patch validation:

1. Installed OS/software and version become package candidates.
2. NVD is searched for CVE candidates.
3. Minimum CVE year and minimum severity settings are applied.
4. Candidate relevance and version evidence are evaluated.
5. Vendor patch sources are checked where supported.
6. Risk and evidence are stored per asset/package/CVE.
7. UI groups rows by unique CVE ID.

Supported validation sources include:

- NVD for general CVE metadata and CVSS.
- Ubuntu Security API and local changelog evidence.
- Debian Security Tracker.
- Red Hat security data.
- Microsoft Security Update Guide/MSRC data for Windows product and KB correlation.
- CISA KEV and public exploit/PoC enrichment.

An NVD API key is strongly recommended. It increases allowed request rate and reduces throttling during large scans.

Vulnerability Statuses

Evidence

Evidence is the authoritative detail. Each row identifies:

- Asset name and instance ID.
- Package and installed version.
- Package manager.
- Risk and CVSS.
- Patch status.
- Fixed version or affected version text where available.
- Microsoft patch state where available.

CLEARED

Patch validation indicates that this asset/package evidence is fixed or not affected by the vulnerable range.

NOT_CLEARED

The evidence remains open, vulnerable, or lacks proof sufficient to mark it cleared.

MIXED

The same CVE has both cleared and open evidence across different assets, packages, or versions. Expand details to identify which evidence still needs action.

Vulnerability tabs

- `All`: every grouped CVE in the current database view.
- `Critical`: critical CVEs with open evidence.
- `Cleared`: CVEs containing cleared evidence; expanded details show cleared evidence for this tab.
- `Actionable`: CVEs with open evidence and risk score of at least 60.

A CVE can appear in both Cleared and Actionable when one asset is patched and another asset remains open. This is expected. Tab-specific expanded evidence is filtered to the relevant assets.

Published date

Each CVE card shows its publication date when provided by the source. `Unknown publish date` means the stored record has no usable date.

Threshold changes

`Minimum Severity` and `Minimum CVE Year` apply when a new scan starts. Changing settings does not rewrite old completed runs. Run a fresh scan to evaluate targets with new thresholds.

Security Group Radar

Security Group Radar combines:

- EC2 security-group attachments from inventory.
- Current ingress rules from AWS EC2.
- Listening ports collected from the target.
- Running service counts from the target.

Summary cards include security groups, assets, public port overlap, public SSH, all-traffic rules, shared rule sets, and best-effort orphan information.

Radar axes:

- `SG`: attached security-group count.
- `Ports`: listening ports.
- `Public`: listener ports overlapping public ingress rules.
- `Svc`: running services.
- `SSH`: public port 22.
- `All`: public all-traffic ingress.

Public exposure recognizes IPv4 `0.0.0.0/0` and IPv6 `::/0` rules.

Interpretation:

- Public security-group permission alone means network policy allows the port.
- A listening port alone means the host has a local listener.
- Public overlap means both were observed and is higher-confidence exposure.
- A zero overlap can still require review when host inventory is missing, stale, or blocked by SSM.

Run inventory refresh and a completed target scan before expecting complete radar data.

Reports

Reports are generated from completed, persisted runs.

Report types

`Target asset workbook`:

- One EC2 target.
- Latest completed run or a selected successful run containing that target.

`Full scan run workbook`:

- All persisted results associated with one completed run.

Download workbook

The full workbook can include:

- Summary.
- EC2/Assets.
- Scan History for asset reports.
- Users or separate Login/Service/System user sheets.
- Groups.
- Ports.
- Services.
- CVEs.
- Critical CVEs.
- Cleared CVEs.
- Actionable CVEs.

Available sheets differ between asset and full-run reports and depend on collected data.

Download Actionable Items report

Creates a reduced workbook using the scanner's actionable classification. Use it as a remediation queue, then verify each row's evidence and patch source before making changes.

Missing report

A report button is disabled when:

- No completed run exists.
- The selected run failed, stopped, or was aborted.
- The selected asset was not included in that run.
- Report metadata was not finalized.

Notifications

The bell contains persistent application notifications:

- New assets.
- Newly tagged assets.
- New critical findings.
- Public SSH or all-traffic exposure.
- CVEs fixed between comparable scans.

Badge colors:

- Red badge on the right: critical/warning notifications.
- Green badge on the left: successful remediation notifications.

A `CVE was fixed` notification requires comparison between two scans of the same target, package manager, package, and CVE. It is not generated from a different target or unrelated package.

`Clear` removes stored notifications. It does not delete scan findings.

Settings

Settings is available only to Admin users.

NVD key

- Stores a custom NVD API key in PostgreSQL.
- Shows only a masked hint after saving.
- Controls vulnerability cache TTL in days.

Cache TTL is the time a package/version lookup can be reused before internet sources are queried again.

Defaults

- Minimum Severity: `NONE`, `LOW`, `MEDIUM`, `HIGH`, or `CRITICAL`.
- Minimum CVE Year.

Defaults are `HIGH` and `2024`. Lower severity or older year settings can increase scan duration and false positives.

Region exclude

Lists enabled AWS regions and accepts comma-separated exclusions.

Example:

```text
ap-south-1, me-south-1
```

Excluded regions are skipped during later inventory refreshes.

Clock settings

- Application timezone.
- 12-hour or 24-hour display.

The selected timezone drives the UI clock and displayed timestamps. The AMI also attempts to update system timezone through the installed helper.

Effects

- Sidebar web animation on/off.
- Current scan eligibility summary.
- Enterprise-only Override Tagging Policy.

Users

Create, rename, change role, change password, deactivate, or reactivate users. At least one active Admin must remain.

Reset actions

All actions require confirmation. Destructive actions also request the current Admin password.

`Force repopulate`:

- Deletes current inventory data and linked asset rows.
- Requires a later inventory refresh to rebuild them.

`Clear all Logs`:

- Clears current live log history from the UI.
- Does not delete completed scan results.

`Force NVD recheck`:

- Removes vulnerability history/cache data.
- A new scan is required to rebuild vulnerability results.

`Fresh Start`:

- Restores scanner data/settings to defaults according to the confirmation text.
- Preserves users and the NVD key.
- Treat as destructive and take a backup first.

Users and Access

Admin

Admins can:

- View all pages.
- Refresh inventory.
- Start, skip, and stop scans.
- Change settings.
- Manage users.
- Run reset actions.

Read-only

Read-only users can:

- View dashboard, assets, scans, vulnerabilities, security groups, and reports.
- Download available reports.

Read-only users cannot:

- Start or control scans.
- Refresh inventory.
- Open Settings.
- Manage users or reset data.

Sessions use an HTTP-only cookie and last up to 14 days unless logged out or invalidated.

Data Storage and Backups

AMI storage model:

- PostgreSQL runs locally.
- PostgreSQL listens on `127.0.0.1:5432`.
- API listens on `127.0.0.1:8000`.
- Nginx exposes HTTPS on `443`.
- Runtime data directory is `/var/lib/spyderz`.
- Scan debug logs are under `/var/lib/spyderz/logs`.
- Build metadata is `/opt/spyderz/BUILD_INFO`.

Persisted in PostgreSQL:

- Workspace settings.
- Users and sessions.
- Assets and inventory snapshots.
- Scan runs and target results.
- Packages, findings, CVE evidence, and patch status.
- Notifications.

Memory-only:

- Current live scan transcript.
- Current inventory-refresh transcript.
- Active worker control state.

Backup

Before upgrades or destructive reset actions:

1. Stop or finish active scans.
2. Create an EBS snapshot of the root volume.
3. For database-level backup, use `pg_dump` while PostgreSQL is healthy.
4. Store backups outside the scanner instance.
5. Test restoration on a separate instance.

Do not rely on AMI replacement as a database backup unless the data volume is also preserved.

Security Hardening

After first login:

1. Change `Admin` password immediately through onboarding.
2. Restrict `443` and `22` to trusted CIDRs or a VPN.
3. Keep `5432` closed and PostgreSQL bound to localhost.
4. Replace the self-signed TLS certificate with a trusted certificate.
5. Use least-privilege IAM and permission boundaries.
6. Restrict SSM SendCommand to intended targets and documents after validation.
7. Use an NVD key dedicated to this scanner.
8. Keep target SSM Agent and the scanner AMI updated.
9. Snapshot data before upgrades.
10. Review active users and deactivate unused accounts.
11. Do not place AWS access keys in files; use the EC2 instance role.
12. Protect SSH private keys and avoid unrestricted `0.0.0.0/0` administration.

The local database password is compiled/configured for application access, but an EC2 administrator with root access controls the host and PostgreSQL. Treat host root access as full access to customer-owned scan data.

Troubleshooting

Basic health check

SSH to the scanner and run:

BASH

```
sudo systemctl status postgresql spyderz-api nginx --no-pager
curl -fsS http://127.0.0.1:8000/api/health
sudo journalctl -u spyderz-api -n 200 --no-pager
sudo ss -ltnp
cat /opt/spyderz/BUILD_INFO
```

Expected listeners:

TEXT

```
127.0.0.1:5432  PostgreSQL
127.0.0.1:8000  Spyderz API
0.0.0.0:443     Nginx HTTPS
```

If PostgreSQL service name differs:

BASH

```
sudo systemctl status postgresql-15 --no-pager
```

UI shows Internal Server Error

Check in this order:

1. `curl http://127.0.0.1:8000/api/health`.
2. `systemctl status postgresql`.
3. `systemctl status spyderz-api`.
4. `journalctl -u spyderz-api`.
5. Free disk space with `df -h`.
6. Memory pressure with `free -h`.

Restart in dependency order:

BASH

```
sudo systemctl restart postgresql
sudo systemctl restart spyderz-api
sudo systemctl restart nginx
```

Do not reset the database before collecting logs and taking a snapshot.

First-login screen repeats

1. Confirm password change returned no error.
2. Log out and log in with the new password.
3. Confirm browser accepts the `spyderz_session` cookie.
4. Check API logs for `/api/auth/complete-onboarding`.
5. Confirm PostgreSQL is writable and has free disk space.
6. Try a private browser window to exclude stale frontend/session state.

No assets appear

Check:

- Scanner IAM allows `ec2:DescribeRegions` and `ec2:DescribeInstances`.
- Region is not excluded.
- Organization SCP does not deny Describe calls.
- Inventory refresh log for `No Access`.
- Target is not terminated.

Asset is visible but not scannable

Check row details:

- State must be `running`.
- SSM must be Online.
- Tag must be exactly `Spyderz=Sense`.
- Enterprise override must be ON if tags are intentionally ignored.
- Standard/Advanced target must fit or already exist in license registry.

On the target, verify SSM Agent:

BASH

```
sudo systemctl status amazon-ssm-agent --no-pager
```

For Windows, inspect the `AmazonSSMAgent` service.

SSM access denied

Check both sides:

- Scanner role: `ssm:DescribeInstanceInformation`, `ssm:SendCommand`, and `ssm:GetCommandInvocation`.

- Target role: `AmazonSSMManagedInstanceCore` or equivalent.
- AWS documents allowed: `AWS-RunShellScript` and `AWS-RunPowerShellScript`.
- SCPs and permission boundaries.
- Region used by the target.

SSM reports no JSON or non-JSON

Possible causes:

- Command timed out.
- PowerShell or Python emitted unexpected text.
- SSM truncated or returned incomplete output.
- Required cmdlet/command is missing.
- Target restarted during collection.
- SSM Agent became offline.

Use the target row message and live transcript to identify the failed section. Windows collection is sectioned, so the last collection message identifies the relevant stage.

Windows shows posture findings but zero CVEs

This can be valid. Check:

- Installed software count in live log.
- OS package line and Windows build.
- NVD key status.
- Minimum CVE year and severity.
- MSRC connectivity.
- Whether accepted matches were cleared by installed KB evidence.

Posture findings such as RDP, SMB, LocalSystem services, or local administrators are not CVEs.

Scan seems stuck

1. Check whether live timestamps are still advancing.
2. Review CPU/MEM indicators.
3. Check API logs and available memory.
4. Identify whether the current stage is NVD or Microsoft patch checking.
5. Wait for the current SSM/API timeout before assuming worker failure.
6. Use Stop Scan once if cancellation is needed.

Weak scanner instances can become unresponsive under collection, CVE matching, report generation, and PostgreSQL load. If CPU/MEM remains red, reduce batch size or resize the scanner before retrying.

Stop Scan does not stop immediately

Stop is checked between logical units. SSM Run Command and external HTTP calls cannot always be interrupted instantly. The UI should stop dispatching new targets, while the current operation returns or times out.

If the API process itself is unhealthy:

```bash
sudo systemctl restart spyderz-api
```

After a process restart, an in-progress stored run can be marked `aborted`.

Vulnerabilities page is empty

Check:

- At least one target scan completed.

- Target collected packages/software.
- Thresholds are not excluding all candidates.
- NVD and vendor APIs are reachable.
- Reported run contains CVE matches, not only posture findings.
- API and PostgreSQL are healthy.

Changing thresholds requires a new scan.

Dashboard charts are empty

Dashboard uses persisted scan data. Complete a scan, then refresh the page. Failed, stopped, or inventory-only operations may not provide enough data for severity split, heat map, packages, or recent CVEs.

Security Group Radar is empty

1. Run inventory refresh.
2. Confirm `ec2:DescribeSecurityGroups`.
3. Complete scans to collect listening ports and services.
4. Check that current asset/security-group attachment rows exist.
5. Review region exclusions and `No Access` region cards.

Reports cannot be downloaded

Only completed successful runs with report metadata are selectable. Confirm:

- Run status is completed.
- Selected asset was included in the run.
- Reports page shows the target under `Assets / With reports`.
- API is healthy.

NVD throttling or slow package matching

- Add a valid NVD API key in Settings.
- Increase cache TTL when repeated scans can reuse recent lookups.
- Scan smaller batches.
- Avoid forcing NVD recheck unless a complete refresh is required.

License registry error

For missing registry:

```TEXT
License registry not found in SSM. Please run setup_license.py or contact support.
```

For invalid HMAC:

```TEXT
License data corrupted or tampered. Contact support.
```

For IAM failure:

```TEXT
AWS IAM setup required.
```

Do not manually repair or recreate a signed registry. Validate IAM first, then use the supplied tier setup/support process.

Known Boundaries

Current scanner intentionally does not perform:

- Full recursive DLL or executable inventory.
- Full file-system hash collection.

- Deep CIS benchmark assessment.
- Domain/GPO analysis.
- Credential or secret scanning.
- Exploit execution.
- Agentless SSH/WinRM collection.

Operational boundaries:

- Live logs and active control state are memory-only.
- Stop is cooperative, not a guaranteed immediate remote-process kill.
- Reports are limited to completed persisted runs.
- Security Group Radar quality depends on both AWS rule access and fresh host listener inventory.
- Vendor APIs can throttle, fail, or return incomplete metadata.
- Package-to-CVE mapping can contain false positives or false negatives, especially for vendor backports and non-standard version strings.
- Microsoft CVE decisions depend on correct OS/product/KB matching.
- Root access to the customer-owned scanner instance implies control over the local database and application runtime.
- Current Standard/Advanced license resolution is region-sensitive.

Always confirm high-impact remediation against vendor advisories and test patches before production deployment.

Support

> For deployment or product support, contact Snakez S.R.O through snakez.sk/operations.html.