

Upskill with StackOps Series

StackOps Resiliency Accelerator (SORA): Discover. Deploy. Observe.

Shan Mahanama, Senior Software Engineer, StackOps

Q & A

Please add your questions for the Q&A session at the end!



Go to
pigeonhole.at

Enter passcode

STACKOPS

Agenda

- 1 SOA Recap** — Where we started — multi-cloud visibility and monitoring automation
- 2 Introducing SORA** — Ground-up redesign with Pulumi, resilience, and a smoother operator experience
- 3 Getting Started** — Install in one command, update in-place, and verify immediately
- 4 What's New** — CLI upgrades: updates, autocomplete, search, themes, and structured error logs
- 5 Kibana Dashboards** — Auto-provisioned OOTB dashboards for AWS across 5 views
- 6 Core Workflow** — check → discover → implement — same shape as SOA, sharper tooling
- 7 Discovery Deep Dive** — 9-stage AWS pipeline and 7-stage Azure pipeline, fully documented
- 8 Safety & Migration** — Pre-flight checks, preview mode, and a clear path from SOA v2 to SORA v3

Recap - SOA

Released in 2024, SOA gave teams complete visibility into their multi-cloud environment — discovering AWS and Azure resources, detecting monitoring gaps, and deploying Elastic Agent infrastructure automatically.

Built on Terraform, it introduced the `soa discover` → Review → `soa implement` workflow that SORA inherits and refines.

```
● ● ● bash

# AWS
$ soa check aws-elastic # 1. Validate credentials + connectivity
$ soa discover aws-elastic # 2. Analyze resources, generate config
$ # 3. Review generated configs
$ soa implement aws-elastic agent # 4. Deploy Elastic Agent infrastructure

# Azure
$ soa check azure-elastic # 1. Validate credentials + connectivity
$ soa discover azure-elastic # 2. Analyze resources, generate config
$ # 3. Review generated configs
$ soa implement azure-elastic agent # 4. Deploy Elastic Agent infrastructure
```

Introduction to SORA

- We gathered feedback from SOA users and built SORA to directly address the pain points they experienced.
- SOA used Terraform for infrastructure management — debugging state issues and recovering from failed runs was complex and time-consuming.
- SORA is a ground-up redesign focused on resilience and simplicity — easier to operate, easier to recover from errors, and more reliable day-to-day.
- All commands in SOA will be available in SORA (some commands are still being migrated).

From SOA → SORA: What Changed

	SOA (v2)	SORA (v3)
Command	soa	sora
Node.js	18+	22+
Infrastructure	Terraform	Pulumi
Discover	Basic pipeline	Multi-stage enriched
AWS Services	Limited	Expanded in v3
Azure Support	Limited	Expanded in v3
Installation	Manual (git clone)	curl install script
Update	Manual reinstall	sora update
Project identifier	--soa.suffix	--project-name
Preview flag	--dry-run	--preview

Getting Started

Install

```
• • • bash
$ curl -fsSL https://download.ops.stack.gov.sg/sora/install.sh | bash
```

Resources

- **Repo:** <https://sgts.gitlab-dedicated.com/innersource/sgts/stackops/automations/stackops-resiliency-accelerator>
- **Docs:** <https://stackops-resiliency-accelerator-e1cc1b.sgts.gitlab-dedicated.site>
- **Migration:** <https://stackops-resiliency-accelerator-e1cc1b.sgts.gitlab-dedicated.site/migration/>

Installation in Action

```
● ● ● bash
# Install
$ curl -fsSL https://download.ops.stack.gov.sg/sora/install.sh | bash
# Verify
$ sora --version
```



Changes

New CLI Experience

- ✓ In-place updates `sora update`
- ✓ Shell autocomplete `sora autocomplete`
- ✓ Command search `sora search`
- ✓ 8 built-in themes `sora config theme`
- ✓ Structured error logs `~/.sora/logs/errors.log`

New Kibana Dashboards

- ✓ OOTB dashboards auto-created on first run
- ✓ Covers overview, resources, packages, containers, risk, and security

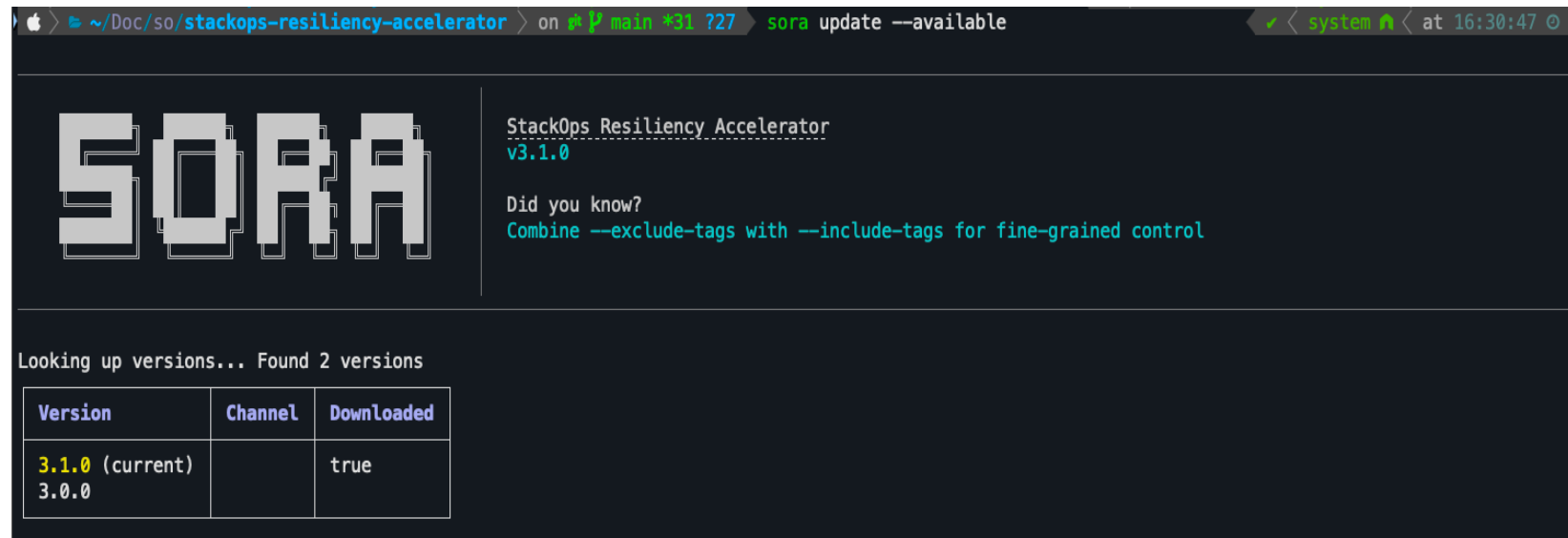
New HTML Report

- ✓ Auto-generated from the discover run — shareable, no terminal needed

In-place Update

Check your installed version and keep SORA current with a single command.

```
# Check version
$ sora --version
# Update to latest
$ sora update
```



```
Apple > ~/Doc/so/stackops-resiliency-accelerator > on P main *31 ?27 sora update --available ✓ < system > at 16:30:47
```



StackOps Resiliency Accelerator
v3.1.0

Did you know?
Combine `--exclude-tags` with `--include-tags` for fine-grained control

Looking up versions... Found 2 versions

Version	Channel	Downloaded
3.1.0 (current)		true
3.0.0		

Improved Tool Output

Stage 1 output from `sora discover aws-elastic`. Each discovered resource shows its type, region, and enrichment status. SORA prints live progress through all 9 pipeline stages.

```

Stage 1/9 - Resource Discovery

i Region: ap-southeast-1 · Account: [REDACTED]

i Resource Explorer Exclusions
  · ec2:snapshot - Transient backup resources
  · ec2:image - Infrastructure templates (AMIs)
  · config:config-rule - AWS Config rules

✓ Registered 19 resource types
→ Discovering all registered types (19 types)
  ✓ [1/5] ec2:instance: 1 resources enriched [0.4s]
  ✓ [2/5] lambda:function: 1 resources enriched [0.3s]
  ✓ [3/5] ecs:service: 1 resources enriched [0.4s]
  ✓ [4/5] apigateway:apis: 1 resources enriched [0.4s]
  ✓ [5/5] elasticloadbalancing:loadbalancer (app): 1 resources enriched [0.4s]
✓ Created 1 synthetic ECR image resources (from ECS container images)

i Active Resource Filters
  · Include tags: sora:project=webinar
  · Exclude tags: sora:ignore=true, soa:filter=exclude, soa:ignore=true

▲ Filtered out 1 resources (based on user-specified tags/name patterns)
i Sample of filtered resources (first 5):
  · public.ecr.aws/docker/library/busybox:latest [ecr:image] tags: NO TAGS

→ Passing 5 resources to next stage

i Summary
  Total Discovered : 6
  Enriched         : 5 (83%)
  Included         : 5 (83%)
  Excluded by filters: 1 (17%)
  
```

Shell autocomplete

Enable tab completion for all SORA commands and flags in your shell.

```
$ sora autocomplete
```

Flags

```

$ sora discover aws-elastic --project-name webinar --include-tags "sora:project-name:webinar"
--agent-type           Agent deployment type to use: "ecs" deploys Elastic Agent on ECS Fargate (default), "ec2" deploys on EC2
--aws-access-key-id    AWS Access Key to be used in the tool.
--aws-region           AWS default region to be used in the tool (only 'ap-southeast-1' is supported now)
--aws-secret-access-key AWS Secret Access Key to be used in the tool.
--aws-session-token    AWS session token to be used in the tool.
--batch-size           Number of resources to process per batch in pipeline stages
--config-output        Output directory for generated aws-elastic-monitoring-configs.json config file.
--debug               -d Enable debug mode. Uses DEBUG env var if set, otherwise enables all namespaces (DEBUG=soa:*). (WARNING: S
--exclude-name-prefix  Exclude resources with names starting with these prefixes. Comma-separated (e.g. internal-,soa-test-)
--exclude-name-suffix  Exclude resources with names ending with these suffixes. Comma-separated (e.g. -test,-dev,-temp)
--exclude-tags         Exclude resources with any of these tags. Multiple key=value pairs separated by comma (e.g. soa:managed=
--full-config          Generate full config with all default values explicitly set. By default a compact config is generated tha
--help                Show help for command
--include-orphaned-logs Include orphaned CloudWatch log groups and S3 buckets (not associated with discovered resources) in the g
--include-tags         Include only resources with at least one of these tags. Overrides exclude-tags. Multiple key=value pairs
--ingestion-mode       Ingestion mode: ha (CloudWatch-S3-SQS, event-driven, default) or standard (direct CloudWatch logs or dire
--kibana-api-key       Kibana API Key to be used in the tool.
--kibana-url           Kibana URL to be used in the tool.
--max-integrations-per-policy Maximum number of log integrations per Fleet agent policy. When exceeded, integrations are split across m
--migrate             Replace soa-managed subscription filters and S3 event notifications with sora-managed ones. Without this
--no-telemetry         Disable telemetry collection and upload for this command invocation.
--project-name         Project name used in generated resource names. Max 12 chars, lowercase alphanumeric + hyphen. Must match:
--report-file          Override the path/filename of the generated HTML monitoring report (aws-elastic-monitoring-report.html).
--sbom-kms-key-arn     Custom KMS key ARN for SBOM S3 encryption (optional, uses S3 default if not set)
--skip-bom-generation Skip BOM (Bill of Materials) generation stage
--skip-elasticsearch-setup Skip Elasticsearch Setup stage (WARNING: Also skips BOM generation and Kibana setup)
--skip-eol-detection   Skip end-of-life (EOL) detection stage
--skip-kibana-setup    Skip Kibana setup (data views, dashboards)
--skip-monitoring-config-detection Skip monitoring configuration detection stage
--skip-package-health-enrichment Skip package health enrichment stage (deprecation and vulnerability data)
--skip-resource-discovery Skip Resource Discovery stage (WARNING: Also skips all dependent stages - SBOM, BOM, vulnerabilities, EOL
--skip-sbom-generation Skip SBOM (Software Bill of Materials) generation stage
--skip-vulnerability-detection Skip vulnerability detection stage
--time-range           Time range to check for data in StackOps/Elastic. Use "m" for minutes, "h" for hours, or "d" for days. e.
--use-direct-api       Use direct AWS API calls instead of Resource Explorer (useful for DynamoDB and other services not indexed

```

Sub-commands

```

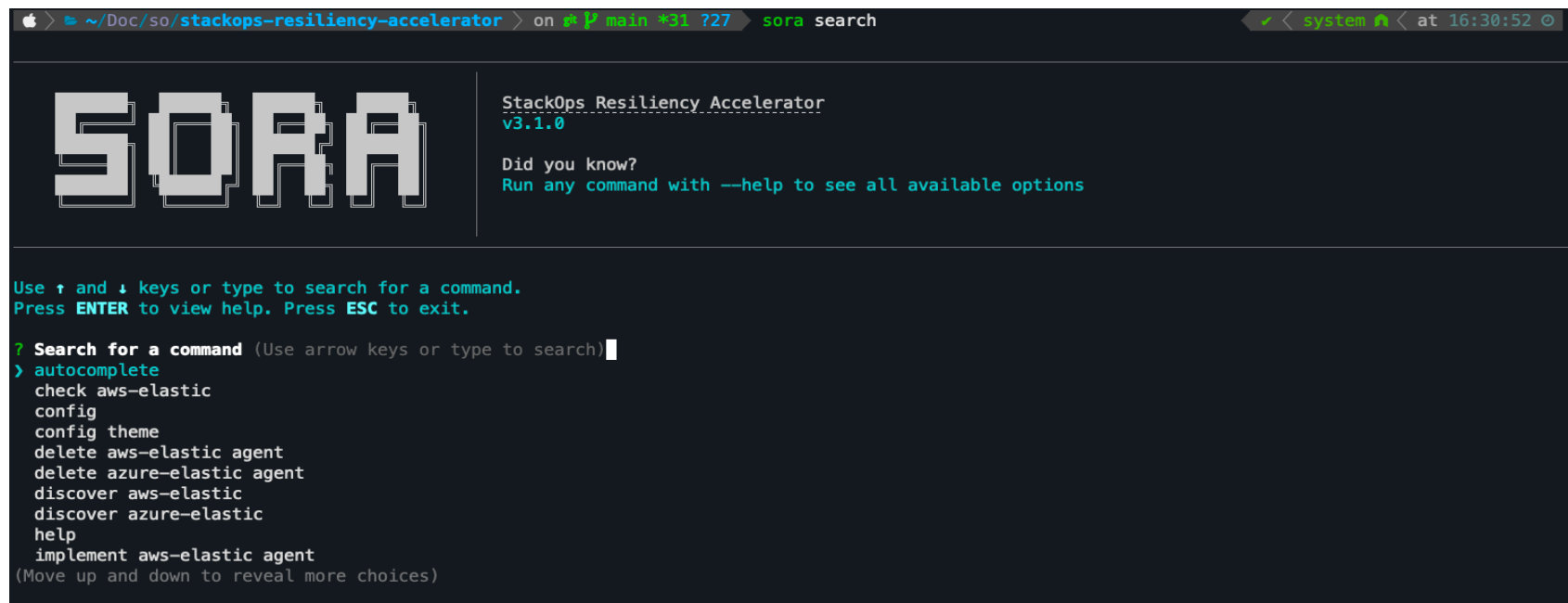
$ sora autocomplete
autocomplete — Display autocomplete installation instructions.
check        — Verify AWS credentials, permissions, and Kibana connectivity before deployment
config       — Configure SORA CLI settings
delete       — Remove Elastic Agent infrastructure from AWS and cleanup Fleet policies in Kibana
discover     — Analyze AWS resources (EC2, ECS, Lambda) and generate configuration file for Elastic Agent deployment
help         — Display help for sora.
implement    — Implement Elastic Agent infrastructure in AWS and configure Fleet policies in Kibana
plugins      — plugins commands
search       — Search for a command.
update       — update the sora CLI
version      —
view         — View available integration configurations
which        — Show which plugin a command is in.

```

Command Search

Search across all SORA commands and subcommands by keyword — find the right command without remembering exact names.

```
$ sora search
```



```
Apple > ~/Doc/so/stackops-resiliency-accelerator > on main *31 ?27 sora search ✓ < system at 16:30:52
```



StackOps Resiliency Accelerator
v3.1.0

Did you know?
Run any command with `--help` to see all available options

Use ↑ and ↓ keys or type to search for a command.
Press ENTER to view help. Press ESC to exit.

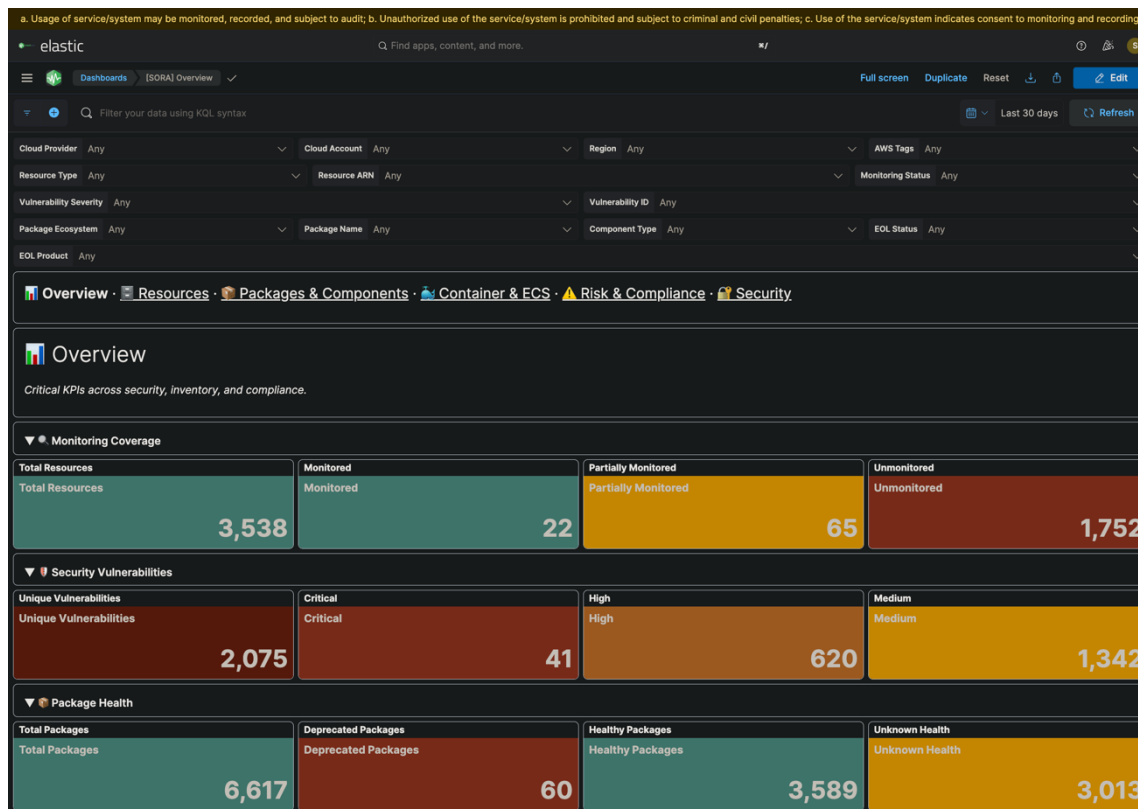
? Search for a command (Use arrow keys or type to search)

- > autocomplete
- check aws-elastic
- config
- config theme
- delete aws-elastic agent
- delete azure-elastic agent
- discover aws-elastic
- discover azure-elastic
- help
- implement aws-elastic agent

(Move up and down to reveal more choices)

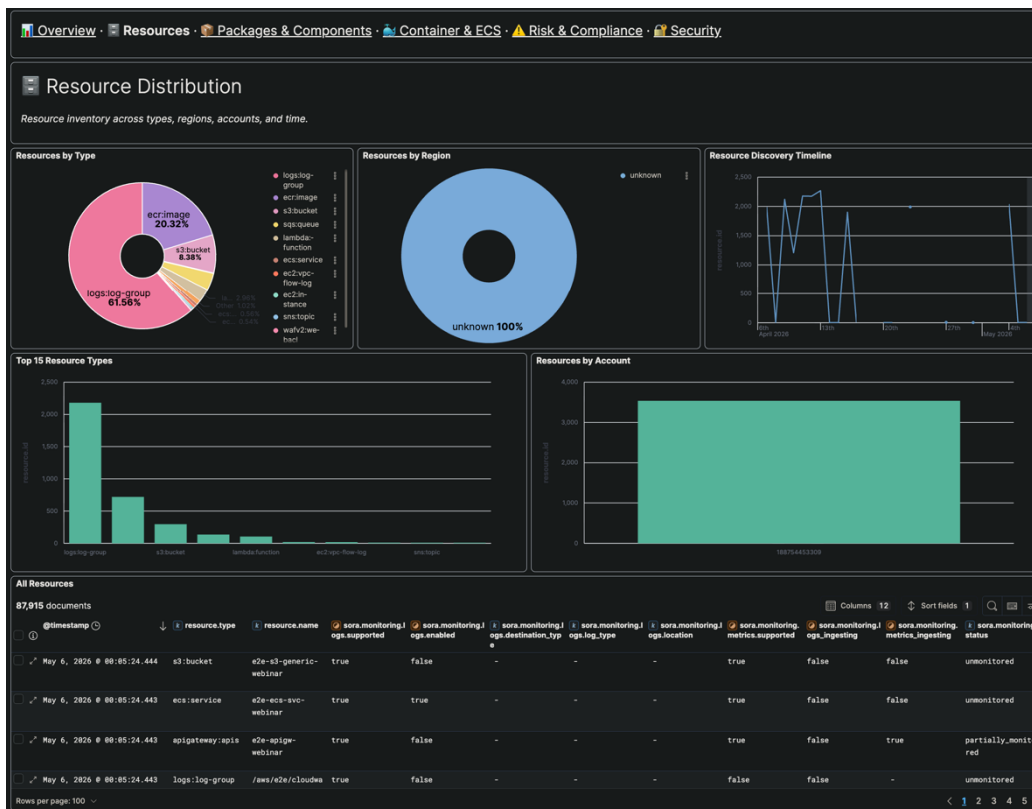
Kibana Dashboard: AWS Overview

KPIs at a glance – monitoring coverage, vulnerability severity counts, and overall package health across all discovered resources.



Kibana Dashboard: AWS Resources

Resource inventory by type, region, and account — discovery timeline and complete resource table.



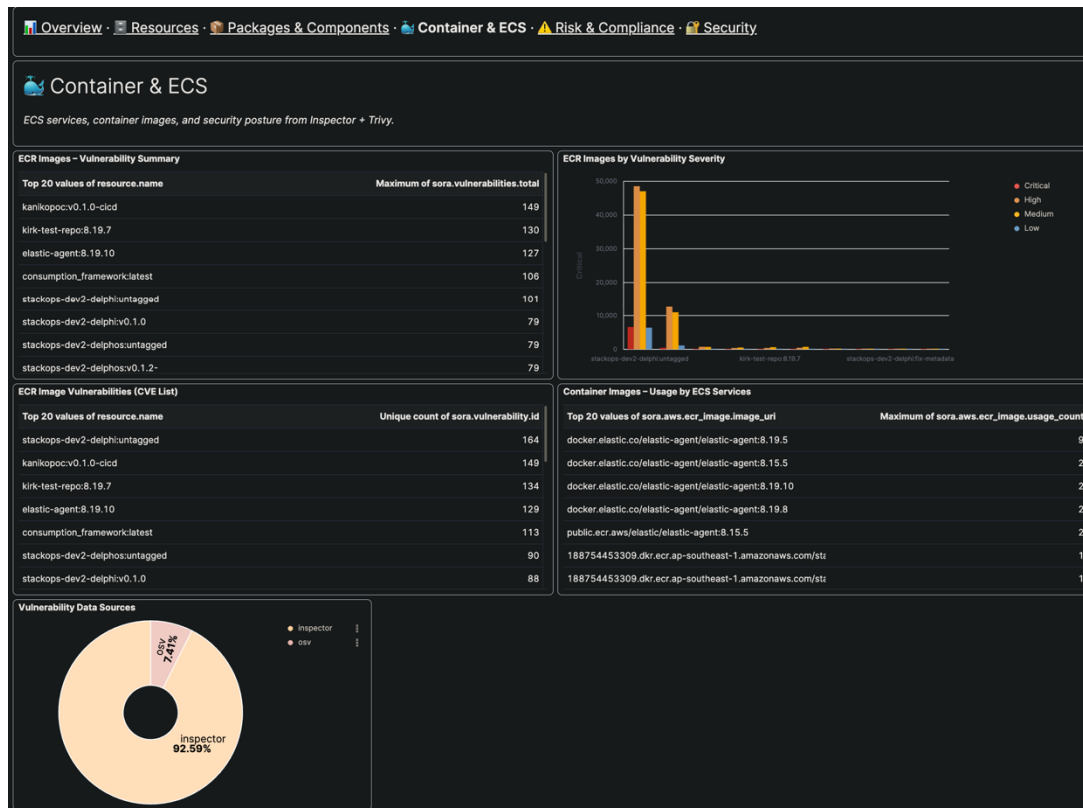
Kibana Dashboard: AWS Packages

Package ecosystem breakdown, component analysis, and SBOM data – top packages by usage and health status.



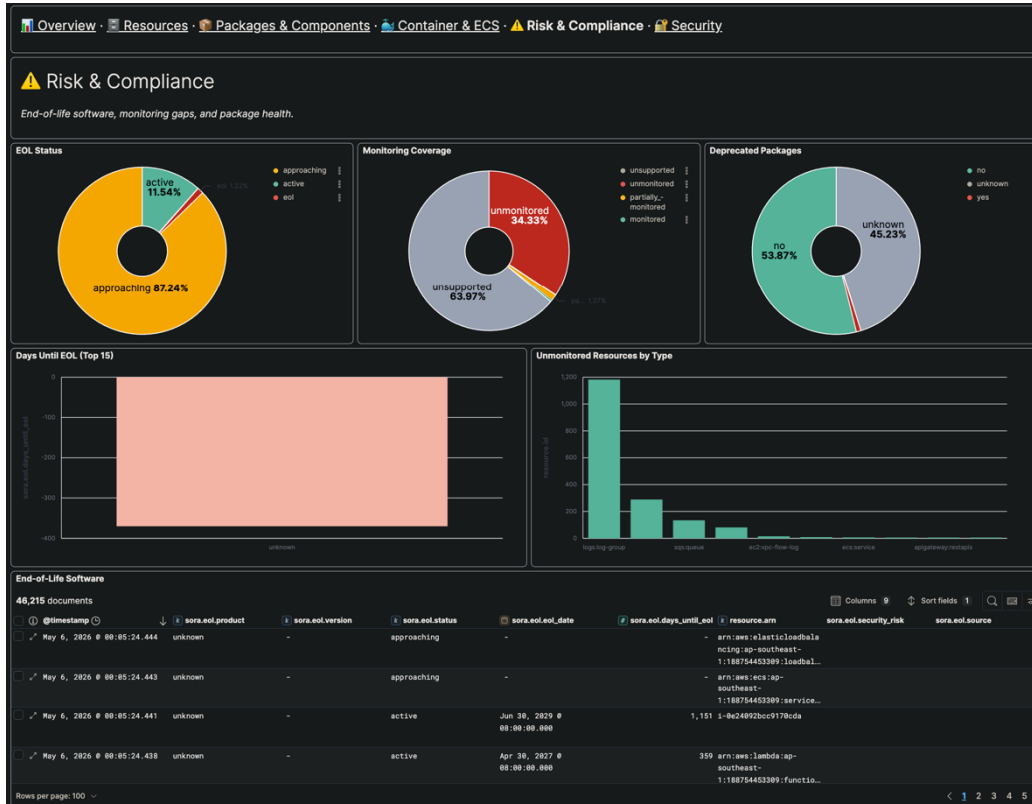
Kibana Dashboard: AWS Containers

ECS services and ECR image security posture – vulnerability findings from AWS Inspector and Trivy, mapped to container usage.



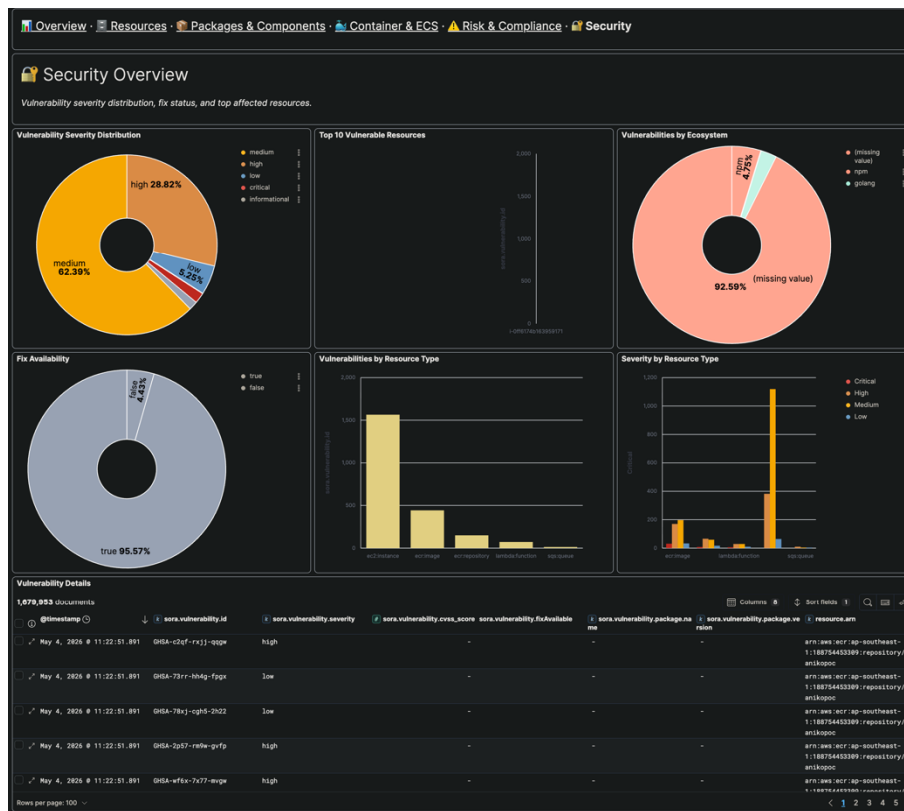
Kibana Dashboard: AWS Risk & Compliance

End-of-life software status, monitoring coverage gaps, and deprecated package tracking across your environment.



Kibana Dashboard: AWS Security

Vulnerability severity distribution, fix availability, and top affected resources — breakdown by ecosystem and resource type.



AWS Monitoring Configuration Report

```

AWS Account ID: 188754453309
ElasticSearch: https://stackopsdevsyste.es.ap-southeast-1.aws.found.io
Time Range: 24h

```

▼ Active Filters (3 active)

Include Tags (must have at least one):

- `sora:project=webinar`

- `soa:filter=exclude`
- `soa:ignore=true`

■ **Include overrides Exclude:** If a resource matches *both* an include tag and an exclude tag, the include tag wins and the resource is kept. Exclude tags only drop resources that have *no* matching include tag.

▼ Resource Breakdown: 7 total (7 processed, 0 excluded, 7 types)

Resource Type	Processed	Excluded	Total	% of Total
apigateway:apis	1	0	1	14.3%
ec2:instance	1	0	1	14.3%
ecs:service	1	0	1	14.3%
elasticloadbalancing:loadbalancer/app	1	0	1	14.3%
lambda:function	1	0	1	14.3%
logs:log-group	1	0	1	14.3%
s3:bucket	1	0	1	14.3%
Total	7	0	7	100.0%

HTML Report (Content)

Auto-generated HTML report from the discover run — full resource inventory, monitoring status, vulnerability findings, and EOL data in a shareable format.

Processed Resources

Resources that passed discovery filters and were successfully enriched with monitoring configuration. Click resource names or log locations to open in AWS Console.

For detailed observability analytics and visualizations, explore the SORA-created [dashboard in Kibana](#).

Tip: Use CMD+Click (Mac) or Ctrl+Click (Windows) on filter items to select multiple values and narrow down results.

Filters Active - 0

Resource Type

Logging Available

Logging Enabled

Log Destination Type

Log Type

Metrics Available

Logs Ingesting (Elastic)

Metrics Ingesting (Elastic)

Overall Monitoring Status

Collapse All

Show All

Clear All

Search:

Resource Type	Resource Name	Logging Available	Logging Enabled	Log Destination Type	Log Type	Log Location (Cloud)	Metrics Available	Logs Ingesting (Elastic)	Metrics Ingesting (Elastic)	Overall Monitoring Status
apigateway:apis	e2e-apigw-webinar	Yes	Yes	CloudWatch	\$default - Access	/aws/apigateway/e2e-apigw-webinar	Yes	No	Yes	PARTIALLY MONITORED
ec2:instance	e2e-ec2-webinar	Yes	Yes	CloudWatch	System Logs	/aws/ec2/i-0e24092bcc9170cda/system	Yes	No	Yes	PARTIALLY MONITORED
ecs:service	e2e-ecs-svc-webinar	Yes	Yes	CloudWatch	app Container Logs	/aws/ecs/e2e-ecs-webinar	Yes	No	No	UNMONITORED
elasticloadbalancing:loadbalancer/app	e2e-elb-webinar	Yes	Yes	S3	Access Logs	s3://e2e-elb-logs-webinar/e2e/elb/webinar	Yes	No	No	UNMONITORED
lambda:function	demo-lambda-webinar	Yes	Yes	CloudWatch	Function Logs	/aws/lambda/demo-lambda-webinar	Yes	No	Yes	PARTIALLY MONITORED
logs:log-group	/aws/e2e/cloudwatch-generic-webinar	Yes	Yes	CloudWatch	Log Group	/aws/e2e/cloudwatch-generic-webinar	N/A	No	N/A	UNMONITORED
s3:bucket	e2e-s3-generic-webinar	Yes	Yes	S3	Server Access Logs	s3://e2e-s3-generic-webinar	Yes	No	No	UNMONITORED

Showing 1 to 7 of 7 entries

Previous

1

Next

Excluded Resources (0 resources)

Generated by SORA v3.1.0.

Core Workflow

```
● ● ● bash

# AWS
$ sora check aws-elastic # 1. Validate credentials + connectivity
$ sora discover aws-elastic # 2. Analyze resources, generate config
$ # 3. Review generated configs
$ sora implement aws-elastic agent # 4. Deploy Elastic Agent infrastructure

# Azure
$ sora check azure-elastic # 1. Validate credentials + connectivity
$ sora discover azure-elastic # 2. Analyze resources, generate config
$ # 3. Review generated configs
$ sora implement azure-elastic agent # 4. Deploy Elastic Agent infrastructure
```



Tip

SORA commands follow the same shape as SOA — only flags have changed. After `sora discover` generates the config file, review and edit it before running `sora implement`.

Demo - Discover



Demo - Implement



Demo - Delete



What Discovery Does For You (AWS)

sora discover aws-elastic runs a 9-stage pipeline:

- 1 **Resource Discovery & Filtering** — EC2, ECS, Lambda — 17 resource types across regions
↓
- 2 **Elasticsearch Setup** — Connection validation and index template preparation
↓
- 3 **Monitoring Config Detection** — Right integration type per discovered resource
↓
- 4 **EOL Detection** — Flags AMIs and OS versions nearing end-of-life
↓
- 5 **Vulnerability Detection** — AWS Inspector + Trivy CVE findings
↓
- 6 **SBOM Generation** — Software Bill of Materials per service (Inspector API)
↓
- 7 **Package Health Enrichment** — npm deprecation and OSV vulnerability checks
↓
- 8 **BOM Generation** — Full resource inventory indexed to Elasticsearch
↓
- 9 **Kibana Setup** — Dashboards and data views provisioned automatically

What Discovery Does For You (Azure)

sora discover azure-elastic runs a 7-stage pipeline:

- 1 **Resource Discovery & Filtering** — *Azure Resource Graph — all supported resource types*
↓
- 2 **Elasticsearch Setup** — *Connection validation and index template preparation*
↓
- 3 **Diagnostic Settings Detection** — *Azure Monitor — what resources already send logs*
↓
- 4 **Subscription Sources Detection** — *Activity Logs and Entra ID audit/sign-in/provisioning logs*
↓
- 5 **Config Generation** — *Produces azure-elastic-monitoring-configs.json + HTML report*
↓
- 6 **BOM Generation** — *Full Azure resource inventory indexed to Elasticsearch*
↓
- 7 **Kibana Setup** — *Dashboards and data views provisioned automatically*

More stages will be added as Azure enrichment (SBOM, vulnerabilities, EOL) rolls out. Current focus is on service discovery, diagnostic configuration, and full BOM coverage.

Azure: Expanded Support

`sora discover azure-elastic` scans your Azure subscription and generates the full monitoring config.

GA:

- Compute — Function Apps, App Services, Virtual Machines (metrics only)
- Networking — Azure Firewall, Application Gateway, Front Door
- Data — CosmosDB, Storage Accounts, Event Hub Namespaces
- AI / Identity — Azure OpenAI / Cognitive Services, Key Vaults
- Platform — Container Registry, Spring Cloud, SQL (MI + DB)

Beta:

- Activity Logs (subscription-level)
- Entra ID — Audit Logs, Sign-in Logs, Identity Protection, Provisioning Logs

Safety Features

Pre-flight check

Validate credentials and connectivity before you commit.

```
● ● ● bash
$ sora check aws-elastic
$ sora check azure-elastic
```

Preview mode

Dry-run inspect before any infrastructure change. No resources are created or modified.

```
● ● ● bash
$ sora implement aws-elastic agent --preview
$ sora delete aws-elastic agent --preview
$ sora implement azure-elastic agent --preview
$ sora delete azure-elastic agent --preview
```

Migrating from SOA v2: *--migrate*

If SOA v2 was managing resources in your AWS environment, those resources may still have SOA-managed configurations attached. SORA won't touch them by default.

Use `--migrate` once during your initial switch to SORA:

```
● ● ● bash
$ sora discover aws-elastic --migrate
$ sora implement aws-elastic agent
```

What it unlocks:

- Log groups at the 2-filter limit with an existing SOA filter → SORA replaces it
- S3 buckets with a SOA-managed notification → SORA takes ownership atomically

After your first successful implement, drop the flag — SORA now owns those resources.

SORA will never migrate resources that have third-party (non-SOA) configurations attached.

Azure users: no migration flag needed — SORA discovers and manages Azure resources fresh.

Planned Roadmap

May 2026

- AWS VPC Endpoint
- Azure Private Endpoint
- AWS Native
- AWS Synthetic

June 2026

- Azure Native
- Azure Synthetic [New]
- AWS Alarms [New]
- Azure Alarms [New]

We are looking for more systems to try cloud-native observability coverage through SORA, please contact us to participate.

Native Baseline: <https://innersource.sgts.gitlab-dedicated.site/sgts/stackops/automations/observability-baseline/>

SOA End of Life: **31 August 2026**

[New] = brand new feature, not available in SOA v2

More AWS and Azure services will be added. Azure resource enrichment (SBOM, vulnerabilities) coming.

Q & A

Please add your questions for the Q&A session !



Go to
pigeonhole.at

Enter passcode

STACKOPS

Let's hear from you!

Share your feedback on the session

Please share your feedback on the session to help us improve in future!

<https://go.gov.sg/stackops-webinar-feedback>



<https://go.gov.sg/stackops-webinar-feedback>

A series of white, wavy, overlapping lines that flow across the top half of the slide, creating a sense of motion and connectivity.

Thank you.