



Supported by:



on the basis of a decision
by the German Bundestag

<https://scs.community/>

CloudLand

CloudLand - 2024-06-12

Make standardization fun again - how we create Sovereign Cloud Stack Standards

Kurt Garloff <garloff@osb-alliance.com>

Dr. Matthias BÜchse <matthias.buechse@cloudandheat.com>

Standardization

Perception

Bureaucracy



Slowness



Political games



SCS Standardization

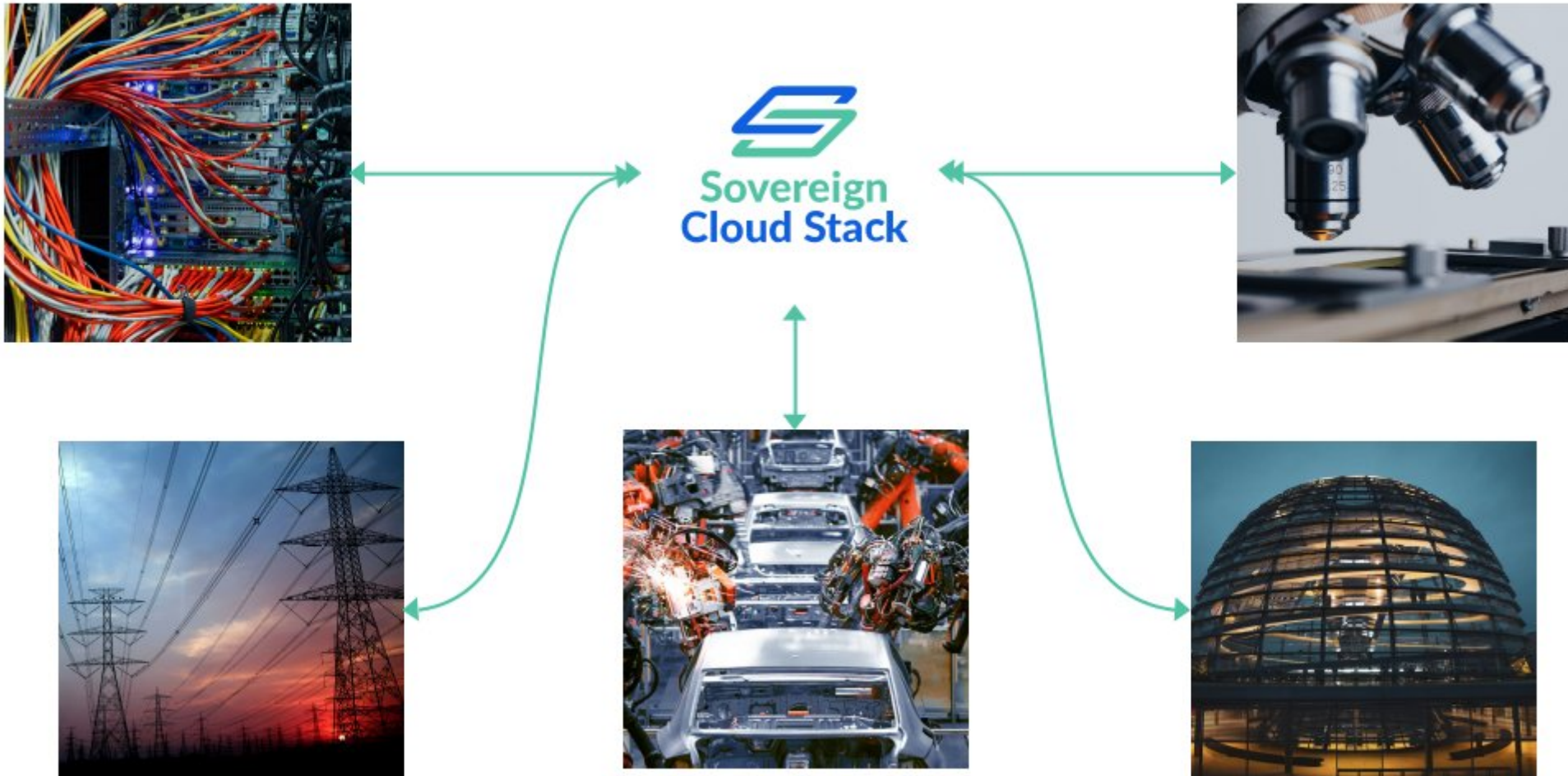
1. We have a mission

One platform - standardized,
built and operated by many.



Sovereign Cloud Stack Mission

Open and federated infrastructure for industry, science, and administration



Digital Sovereignty & SCS Certification

Levels of digital sovereignty

4: Operational Transparency and accessible Knowledge (Skill building)

3: Technological Transparency and ability to contribute and shape

2: Choice between many operators, insourcing option (on-premise)

1: Compliance with regulation (GDPR)



SCS Certification Level



4: “SCS-sovereign” – Ops/IAM stacks are OSS; transparency on monitoring and incidents, contribution to Open Operations (5 Opens)

3: “SCS-open” – SBOM for functional stack available and all openly developed OSS (4 Opens)

2: “SCS-compatible” – technical compatibility (conformity tests from CNCF, OIF, SCS)

1: (Not SCS-specific): ENISA/Gaia-X labels & legal rules

Sovereign Cloud Stack Deliverables



1

Certifiable Standards



2

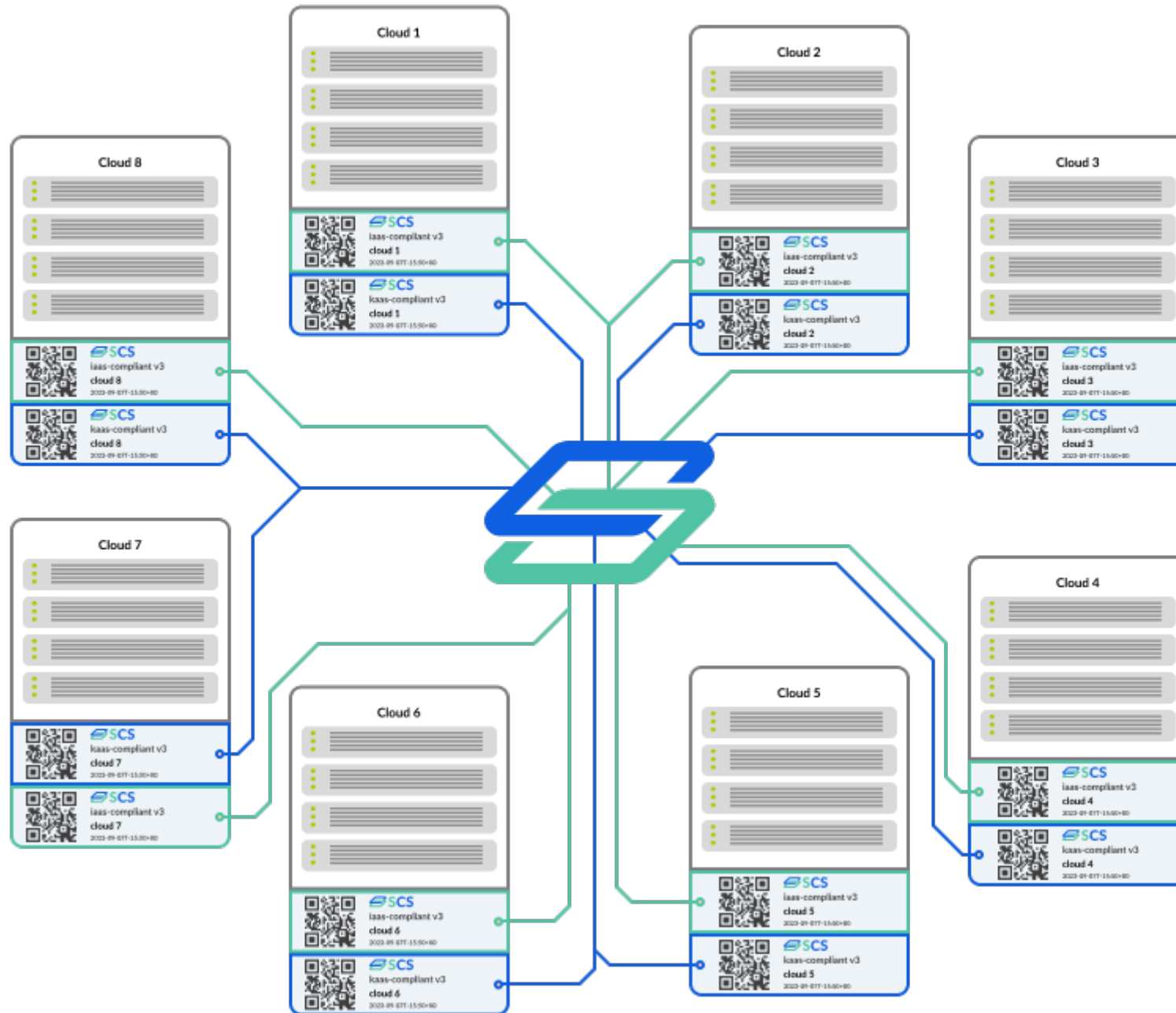
Modular Open Source
Reference Implementation



3

Accessible
operational knowledge

SCS = an open, federated, virtual Hyperscaler



Federate
public,
private, &
community
SCS clouds

- standards
- IAM
- networked

SCS Standardization

2. We treat it like Open Source Software development

Open, community-driven process



Discussions w/
upstream and with
CSPs & customers

Github:

- Issues
- PRs w/ reviews

Bi-Weekly
VideoConf
(on SCS public
calendar)

- For non-obvious
decisions

Process description

Files

main + 🔍

Go to file t

- > .github
- > .husky
- > .vscode
- > .zuul.d
- > Drafts
- > Operational-Docs
- ✓ Standards
 - README.md
 - scs-0001-v1-sovereign-cloud-sta...
 - scs-0002-v1-standards-docs-org....
 - scs-0002-v2-standards-docs-org....
 - scs-0003-v1-sovereign-cloud-sta...
 - scs-0004-v1-achieving-certificati...
 - scs-0100-v1-flavor-naming.md
 - scs-0100-v2-flavor-naming.md
 - scs-0100-v3-flavor-naming.md
 - scs-0100-w1-flavor-naming-impl...
 - scs-0101-v1-entropy.md
 - scs-0101-w1-entropy-implemen...
 - scs-0102-v1-image-metadata.md
 - scs-0103-v1-standard-flavors.md
 - scs-0104-v1-standard-images.md
 - scs-0110-v1-ssd-flavors.md
 - scs-0111-v1-volume-type-decisi...
 - scs-0112-v1-conic.md

standards / Standards / scs-0001-v1-sovereign-cloud-standards.md ↑ Top

Preview Code Blame 376 Lines (276 loc) · 15.6 KB Raw 📄 ⬇️ ✎ ⋮

- A section containing the actual decision that is introduced. The section should also include reasoning for this decision.

We also RECOMMEND the following sections:

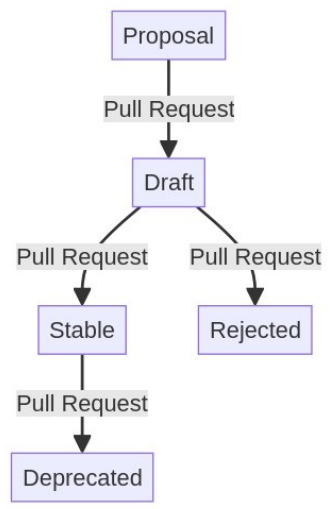
- A *Terminology* section which shortly describes terms used in the document, including possible abbreviations.
- A *Related Documents* section which references related Standards or Decisions, both upstream and/or other SCS documents.

In addition, the following OPTIONAL sections should be considered:

- A *Consequences* section describing outcomes from implementing the changes described.

Process

The lifecycle of an SCS document goes through the following phases: Proposal, Draft, Stable, Deprecated, and Rejected.



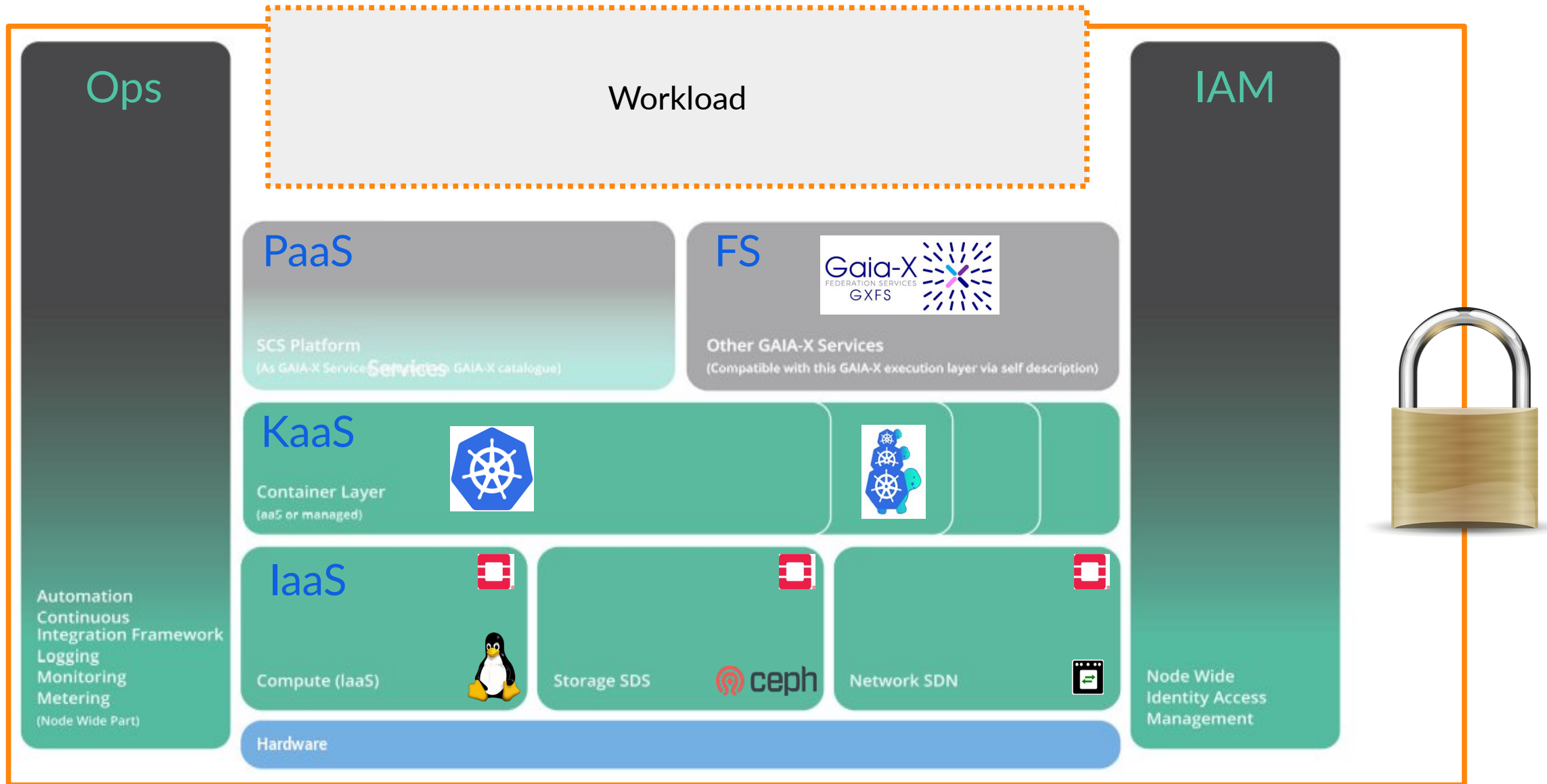
All decisions for phase transitions follow loose consensus, where the group which has to form the consensus depends on the track of the document:

SCS Standardization

3. Sync with Reference Implementation

SCS Architecture (Software/Ref.Impl.)

Standardization aligned with Ref.Impl. Architecture



SCS Platform Services (PaaS) are planned
 Hardware and Federation Services not part of SCS software
 KaaS = Kubernetes as a Service

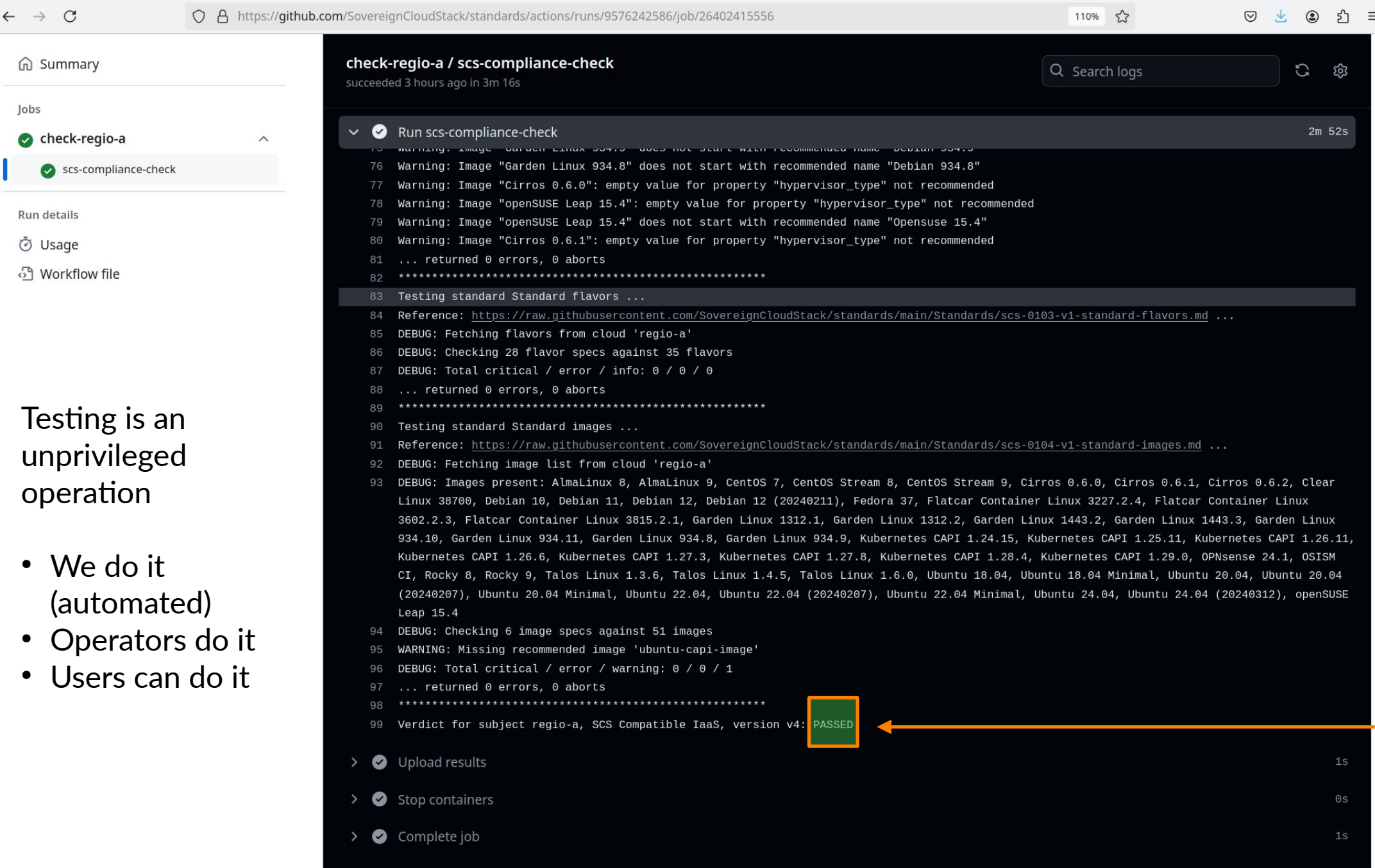
Implementation experts = Standardization experts



SCS Standardization

4. Compliance test development

Continuous compliance check (github)



The screenshot shows a GitHub Actions workflow run for the job 'check-regio-a / scs-compliance-check'. The run is successful and completed 3 hours ago in 3m 16s. The main task 'Run scs-compliance-check' is highlighted and shows a 'PASSED' status. The log output includes several warning messages about image names and empty values for 'hypervisor_type', followed by a 'Verdict for subject regio-a, SCS Compatible IaaS, version v4: PASSED'.

```
75 warning: Image "Garden Linux 934.9" does not start with recommended name "Debian 934.9"
76 Warning: Image "Garden Linux 934.8" does not start with recommended name "Debian 934.8"
77 Warning: Image "Cirros 0.6.0": empty value for property "hypervisor_type" not recommended
78 Warning: Image "openSUSE Leap 15.4": empty value for property "hypervisor_type" not recommended
79 Warning: Image "openSUSE Leap 15.4" does not start with recommended name "Opensuse 15.4"
80 Warning: Image "Cirros 0.6.1": empty value for property "hypervisor_type" not recommended
81 ... returned 0 errors, 0 aborts
82 *****
83 Testing standard Standard flavors ...
84 Reference: https://raw.githubusercontent.com/SovereignCloudStack/standards/main/Standards/scs-0103-v1-standard-flavors.md ...
85 DEBUG: Fetching flavors from cloud 'regio-a'
86 DEBUG: Checking 28 flavor specs against 35 flavors
87 DEBUG: Total critical / error / info: 0 / 0 / 0
88 ... returned 0 errors, 0 aborts
89 *****
90 Testing standard Standard images ...
91 Reference: https://raw.githubusercontent.com/SovereignCloudStack/standards/main/Standards/scs-0104-v1-standard-images.md ...
92 DEBUG: Fetching image list from cloud 'regio-a'
93 DEBUG: Images present: AlmaLinux 8, AlmaLinux 9, CentOS 7, CentOS Stream 8, CentOS Stream 9, Cirros 0.6.0, Cirros 0.6.1, Cirros 0.6.2, Clear
Linux 38700, Debian 10, Debian 11, Debian 12, Debian 12 (20240211), Fedora 37, Flatcar Container Linux 3227.2.4, Flatcar Container Linux
3602.2.3, Flatcar Container Linux 3815.2.1, Garden Linux 1312.1, Garden Linux 1312.2, Garden Linux 1443.2, Garden Linux 1443.3, Garden Linux
934.10, Garden Linux 934.11, Garden Linux 934.8, Garden Linux 934.9, Kubernetes CAPI 1.24.15, Kubernetes CAPI 1.25.11, Kubernetes CAPI 1.26.11,
Kubernetes CAPI 1.26.6, Kubernetes CAPI 1.27.3, Kubernetes CAPI 1.27.8, Kubernetes CAPI 1.28.4, Kubernetes CAPI 1.29.0, OPNsense 24.1, OSISM
CI, Rocky 8, Rocky 9, Talos Linux 1.3.6, Talos Linux 1.4.5, Talos Linux 1.6.0, Ubuntu 18.04, Ubuntu 18.04 Minimal, Ubuntu 20.04, Ubuntu 20.04
(20240207), Ubuntu 20.04 Minimal, Ubuntu 22.04, Ubuntu 22.04 (20240207), Ubuntu 22.04 Minimal, Ubuntu 24.04, Ubuntu 24.04 (20240312), openSUSE
Leap 15.4
94 DEBUG: Checking 6 image specs against 51 images
95 WARNING: Missing recommended image 'ubuntu-capi-image'
96 DEBUG: Total critical / error / warning: 0 / 0 / 1
97 ... returned 0 errors, 0 aborts
98 *****
99 Verdict for subject regio-a, SCS Compatible IaaS, version v4: PASSED
```

Testing is an unprivileged operation

- We do it (automated)
- Operators do it
- Users can do it

PASSED

Continuous compliance check (zuul)

← → ↻ https://zuul.scs.community/t/SCS/builds?job_name=scs-check-poc-wgcloud&project=SovereignCloudStack/standards ☆ ⌵ ⌵ ⌵ ⌵ ⌵

ZUUL Components API Documentation Tenant SCS UTC

Status Projects Jobs Labels Nodes Autoholds Semaphores **Builds** Buildsets

Job Filter by Job... 🔍

Job scs-check-poc-wglo... Project SovereignCloudStack... Clear all filters

1 - 5 of 5 < >

Job	Project	Branch	Pipeline	Change	Duration	Start time	Result
✓ scs-check-poc-wgcloud	SovereignCloudStack/standards	main	periodic-daily		6 mins 35 secs	2024-06-19 03:13:50	SUCCESS
✓ scs-check-poc-wgcloud	SovereignCloudStack/standards	main	periodic-daily		7 mins 2 secs	2024-06-18 03:07:10	SUCCESS
✓ scs-check-poc-wgcloud	SovereignCloudStack/standards	main	periodic-daily		17 mins 10 secs	2024-06-17 03:08:58	SUCCESS
! scs-check-poc-wgcloud	SovereignCloudStack/standards	main	periodic-daily		0 secs	2024-06-16 03:10:56	NODE_FAILURE
! scs-check-poc-wgcloud	SovereignCloudStack/standards	main	periodic-daily		0 secs	2024-06-15 03:08:10	NODE_FAILURE

SCS Standardization

Results



- Introduction
- Certification ▼
- Scopes and Versions ▼
 - SCS Compatible IaaS
 - SCS Compatible KaaS
- Standards** ▼
- Global ▼
 - scs-0001 >
 - scs-0002 >
 - scs-0003 >
 - scs-0004 >
 - scs-0112 >
- IaaS ▼
 - scs-0100 >
 - scs-0101 >
 - scs-0102 >
 - scs-0103 >
 - scs-0104 >
 - scs-0110 >
 - scs-0111 >
- KaaS ▼
 - scs-0200 >
 - scs-0210 >
 - scs-0211 >
 - scs-0212 >
 - scs-0213 >

🏠 > Standards

Overview

Standards are the core deliverable of SCS. By standardizing the open source software components of a cloud computing stack, their versions, how they are to be configured, deployed and utilized, SCS guarantees the reproducibility of a certain behavior of this technology.

SCS standards are discussed, developed and maintained in the community by the corresponding teams (see Track in the table below), which naturally include existing users of SCS.

*Legend to the column headings: Draft, Stable (but not effective), Effective, Deprecated (and no longer effective).

Standard	Track	Description	Draft	Stable*	Effective	Deprecated*
scs-0001	Global	Sovereign Cloud Standards	-	-	v1	-
scs-0002	Global	Standards, Docs and Organisation	v2	-	v1	-
scs-0003	Global	Sovereign Cloud Standards YAML	v1	-	-	-
scs-0004	Global	Regulations for achieving SCS-compatible certification	v1	-	-	-
scs-0112	Global	SONiC Support in SCS	v1	-	-	-
scs-0100	IaaS	SCS Flavor Naming Standard	-	-	v3	v1, v2
		Supplement: Implementation and Testing Notes	w1	-	-	-
scs-0101	IaaS	SCS Entropy	-	-	v1	-
		Supplement: Implementation and Testing Notes	w1	-	-	-
scs-0102	IaaS	SCS Image Metadata Standard	-	-	v1	-
scs-0103	IaaS	SCS Standard Flavors and Properties	-	-	v1	-

SCS standards: IaaS

Goals:

- Make IaC tooling (Terraform/OpenTofu, Ansible, OpenStack SDK+CLI) work across SCS IaaS
- Ensure SCS Cluster Stacks (CAPO w/ Cluster Classes and Cluster-Stack Operator) work across SCS IaaS
- Ensure Security and QoS

Topics:

- Base on upstream OpenStack InterOp work (OpenStack powered Compute) ✓✓
- Flavor naming, properties, standard flavors ✓✓✓
- Images, image handling, metadata ✓✓✓
- VM Entropy ✓✓✓
- Virtual network model (local networks, external network access, security groups) (✓)
- DNS and NTP service (✓)
- Domain-manager persona (self-service capability) ✓
- Volume types (redundancy, encryption, performance) ✓
- Loadbalancer availability and requirements (✓)
- Meaning of AZ and Regions (✓)

Legend:

✓✓✓: Standardized, required, tested

✓✓: Standardized, required

✓: ADR/ Draft standards

(✓): Only in ref. implementation

SCS standards: KaaS

Goals:

- Ensure k8s clusters behave in defined ways
- Ensure Security and QoS

Topics:

- Base on upstream CNCF E2E conformance tests ✓✓✓
- K8s patch availability (version recency) ✓✓✓
- CNI (w/ network policies) (✓)
- CSI (storage classes and properties) (✓)
- LoadBalancer (for optional ingress/gateway API) (✓)
- Node distribution (anti-affinity) ✓✓
- Metrics service (opt-out) (✓)
- Container registry (opt-in) (✓)
- Cluster-Management / GitOps => [Separate KaaS Cluster Mgmt Standard](#)

SCS standards: Ops, IAM

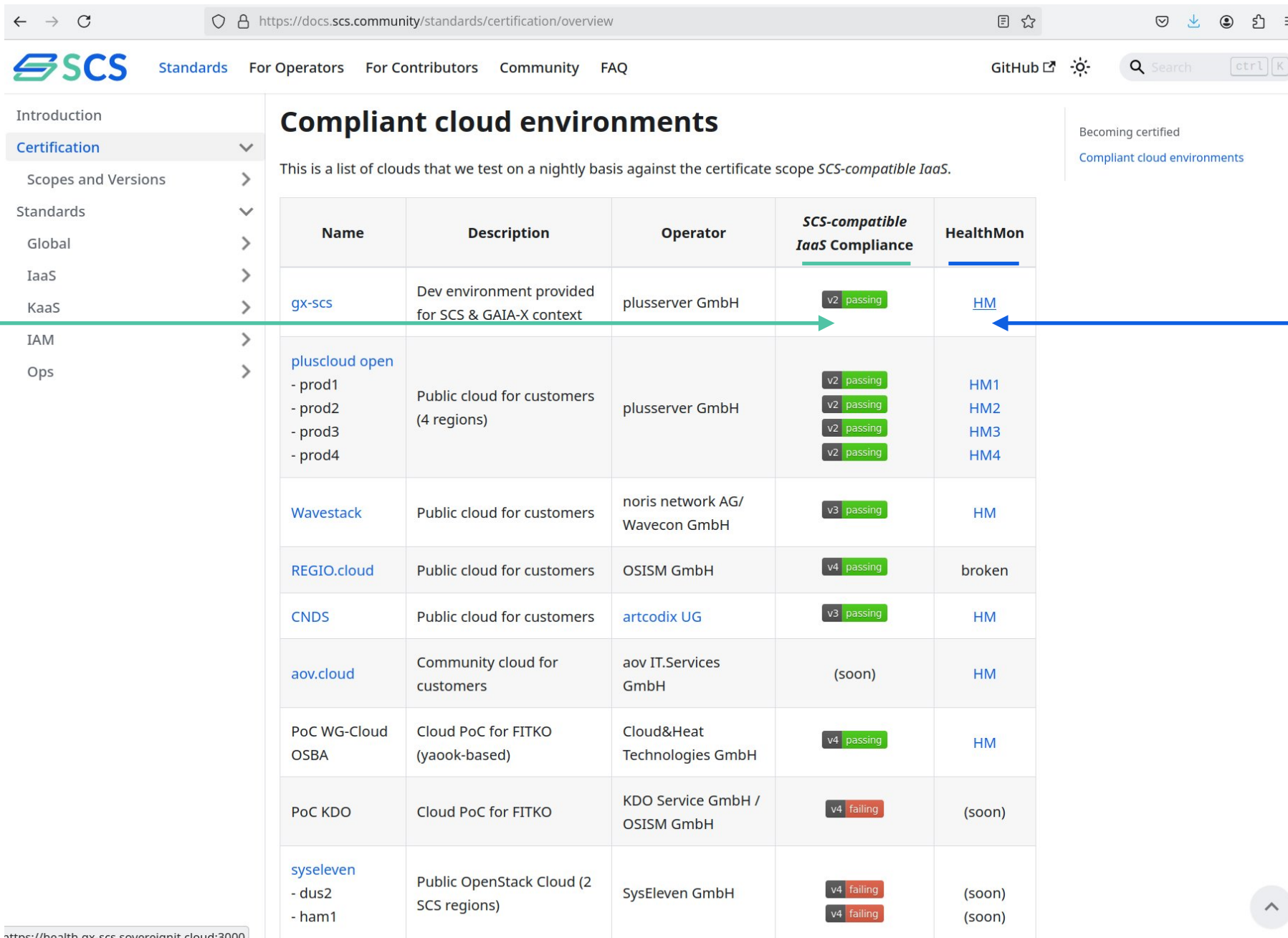
Goals:

- Ensure transparency over Operational Properties
- Ensure federatability

Topics:

- Health status (both IaaS and KaaS) (✓)
- Status page (✓)
- Identity federation IaaS & KaaS → ID Broker (OIDC) (✓)
- Identity federation Broker → External (OIDC/SAML) (✓)

Standard conformity and transparency



Introduction

Certification ▾

- Scopes and Versions >
- Standards ▾
- Global >
- IaaS >
- KaaS >
- IAM >
- Ops >

Compliant cloud environments

This is a list of clouds that we test on a nightly basis against the certificate scope *SCS-compatible IaaS*.

Name	Description	Operator	SCS-compatible IaaS Compliance	HealthMon
gx-scs	Dev environment provided for SCS & GAIA-X context	plusserver GmbH	v2 passing	HM
pluscloud open	Public cloud for customers (4 regions)	plusserver GmbH	v2 passing	HM1
- prod1			v2 passing	HM2
- prod2			v2 passing	HM3
- prod3			v2 passing	HM4
Wavestack	Public cloud for customers	noris network AG/ Wavecon GmbH	v3 passing	HM
REGIO.cloud	Public cloud for customers	OSISM GmbH	v4 passing	broken
CNDS	Public cloud for customers	artcodix UG	v3 passing	HM
aov.cloud	Community cloud for customers	aov IT.Services GmbH	(soon)	HM
PoC WG-Cloud OSBA	Cloud PoC for FITKO (yaook-based)	Cloud&Heat Technologies GmbH	v4 passing	HM
PoC KDO	Cloud PoC for FITKO	KDO Service GmbH / OSISM GmbH	v4 failing	(soon)
sysleven	Public OpenStack Cloud (2 SCS regions)	SysEleven GmbH	v4 failing	(soon)
- dus2			v4 failing	(soon)
- ham1				

[Becoming certified](#)

[Compliant cloud environments](#)

<https://health.gx-scs.sovereignit.cloud:3000>

Daily updated standard conformity result (here: IaaS SCS-compatible)

OpenStack Health Monitor dashboard: Public realtime monitoring of errors and performance



Adoption

https://docs.scs.community/standards/certification/overview

SCS Standards For Operators For Contributors Community FAQ

Introduction Certification Scopes and Versions Standards Global IaaS KaaS IAM Ops

Compliant cloud environments

This is a list of clouds that we test on a nightly basis against the certificate scope *SCS-compatible IaaS*.

Name	Description	Operator	SCS-compatible IaaS Compliance	HealthMon
gx-scs	Dev environment provided for SCS & GAIA-X context	plusserver GmbH	v2 passing	HM
pluscloud open	Public cloud for customers (4 regions)	plusserver GmbH	v2 passing	HM1
- prod1			v2 passing	HM2
- prod2			v2 passing	HM3
- prod3			v2 passing	HM4
Wavestack	Public cloud for customers	noris network AG/ Wavecon GmbH	v3 passing	HM
REGIO.cloud	Public cloud for customers	OSISM GmbH	v4 passing	broken
CNDS	Public cloud for customers	artcodix UG	v3 passing	HM
aov.cloud	Community cloud for customers	aov IT.Services GmbH	(soon)	HM
PoC WG-Cloud OSBA	Cloud PoC for FITKO (yaook-based)	Cloud&Heat Technologies GmbH	v4 passing	HM
PoC KDO	Cloud PoC for FITKO	KDO Service GmbH / OSISM GmbH	v4 failing	(soon)
syseseven	Public OpenStack Cloud (2 SCS regions)	SysEleven GmbH	v4 failing	(soon)
- dus2			v4 failing	(soon)
- ham1				

https://health.gx-scs.sovereignit.cloud:3000

Becoming certified
[Compliant cloud environments](#)

OpenStack IaaS using SCS reference implementation

OpenStack IaaS using independent implementation

SCS standards used in DVC, S-H, ZenDiS, ...

FITKO PoC (OSBA) for provider switch

OSBA Forum SCS Standardization

Considerations behind SCS standards in govstack UN initiative

SCS Summit May 2024



Supporting companies and organizations

				Upcoming:

Existing SCS operator

Future SCS operator

SCS compat

Impl. Partner



— An OSB ALLIANCE project —

<https://scs.community/>



<https://scs.community>

Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

