Craig Simpson CV

2026-01-16

Contents

pdf-test

GitHub Actions test to automatically generate C.V Updates.

PDF publishing workflow

This repo includes a GitHub Actions workflow that converts `docs/DISCOVERY.md` into a PDF and commits it to another repository.

Configuration (edit in `.github/workflows/publish-pdf.yml`): - `DEST_REPO`: target repository in `owner/name` form. - `DEST_BRANCH`: target branch (default `main`). - `DEST_PATH`: folder in the target repo to receive the PDF. - `DEST_FILE`: PDF filename in the target repo.

Secrets required in this repo: - `DEST_REPO_TOKEN`: a PAT with write access to the destination repository. title

OpenShift Discovery Plan

Document status: Draft
Last updated: 2026-01-13
Owner:
Audience: Platform Engineering, Networking, Security, App Teams

---

1. Purpose

This document captures the discovery inputs required to design, deploy, and operate an OpenShift platform in a consistent and supportable way.

Goals - Establish a shared understanding of current-state infrastructure and constraints - Identify dependencies, owners, and points of contact - Reduce delivery risk by surfacing unknowns early

Non-goals - Detailed implementation runbooks (tracked separately) - Application-by-application migration plan (linked in Appendix)

---

2. Scope

In scope - Platform topology (on-prem / cloud / hybrid) - Network and DNS design inputs - Hardware and ownership - Restricted networks and external connectivity - Supporting tools (CI/CD, registry, version control)

Out of scope - End-user training - FinOps cost optimisation beyond initial sizing

---

3. Current State Summary

Area	Current state	Notes / gaps
Compute		<e.g., lifecycle constraints>
Storage		<IOPS targets?>
Network		<segmentation model?>
Identity		<group strategy?>
Monitoring		<ownership?>

---

4. Platform Topology

4.1 Environments

- **Dev:**
- **Test:**
- **Prod:**

4.2 Cluster count and purpose

- Cluster 1: —
- Cluster 2: —

4.3 Availability targets

- Control plane:
- Worker pools:
- RTO/RPO:

---

5. Networking

5.1 Network overview

- **Pod CIDR:** `<x.x.x.x/xx>`
- **Service CIDR:** `<x.x.x.x/xx>`
- **Machine network(s):** `<x.x.x.x/xx>`
- **Egress model:** `<NAT / proxies / direct>`
- **Ingress model:** `<routes / LB / WAF>`

5.2 Firewalls and segmentation

- Key north/south boundaries:
- East/west restrictions:
- Required openings (initial):
  - `<src>` → `<dst>` : `<ports/protocols>` : `<reason>`

5.3 Load balancers

- API LB: , VIP: <x.x.x.x>
- Ingress LB: , VIPs: <x.x.x.x>
- Health-check method: , endpoints:

5.4 VPNs, Private Links, and Data Center Interconnects

- VPNs in use:
- Private links: <AWS/Azure/GCP private endpoints, etc.>
- DCI: , bandwidth: , latency:
- Routing responsibility:

---

6. DNS

6.1 Local domains

- Cluster base domain: `<apps.example.internal>`
- Corporate internal domain(s): `<example.internal>`, `<corp.example.com>`

6.2 Name servers

Authoritative DNS - Primary: `<ns1.example.internal>` — Owner: - Secondary: `<ns2.example.internal>` — Owner:

Recursive DNS - Resolver 1: `<resolver1.example.internal>` — Owner: - Resolver 2: `<resolver2.example.internal>` — Owner:

6.3 Records required (examples)

- `api.<cluster>.<domain>` → `<vip>`
- `api-int.<cluster>.<domain>` → `<vip>`
- `*.apps.<cluster>.<domain>` → `<ingress vip>`

---

7. Restricted Networks and External Connectivity

7.1 Restricted network definition

- Internet access from cluster nodes: `<none / limited / via proxy>`
- Allowed outbound destinations:
- TLS inspection: `<yes/no>`, exceptions:

7.2 External bridges / cloud connectivity

- Bridge to **Microsoft Azure** (for ARO / hybrid):
  - Connectivity type: `<ExpressRoute / VPN / Peering>`
  - Egress points: `<details>`
  - Constraints: `<e.g., no public endpoints>`

7.3 Proxy configuration (if applicable)

- HTTP proxy: `<http://proxy:port>`
- HTTPS proxy: `<http://proxy:port>`
- No-proxy: `<.cluster.local,.internal,10.0.0.0/8,...>`

---

8. Hardware Details

Include what is deployed, how it's configured, and who to contact.

8.1 Inventory

Component	Qty	Model / SKU	Location	Lifecycle	Notes
Control plane nodes					
Worker nodes					
Storage					

8.2 Configuration standards

- BIOS/firmware baseline:
- NIC bonding/VLAN model:
- Time sync (NTP):
- Out-of-band access: , network:

8.3 Points of contact

- Hardware operations: ,
- Data center: ,

---

9. Ownership and Contacts

Area	Owner team	Primary contact	Backup contact
Red Hat software (OpenShift)			
Networking			
Hardware			
Security			
Identity			

---

10. Ancillary Services

10.1 Version control

- System:
- Org/project structure:
- Access model:

10.2 CI/CD

- Tools:
- Promotion strategy: <dev→test→prod>
- Secrets management:

10.3 Container registry

- Registry:
- Image scanning:
- Retention policy:

---

11. Risks and Open Questions

Top risks 1. — Impact: — Mitigation: 2. — Impact: — Mitigation:

Open questions - [ ] - [ ]

---

12. Decisions Log

Date	Decision	Rationale	Owner
2026-01-13			

---

13. Next Steps

- ☐ Confirm network CIDRs and firewall openings
- ☐ Validate DNS ownership and required records
- ☐ Finalise hardware sizing and lifecycle constraints
- ☐ Confirm ancillary tooling integration approach
- ☐ Schedule design review and sign-off

Appendix A: Links

- Architecture diagram:
- Firewall request tracker:
- Hardware inventory source:
- CI/CD standards: