Kubernetes in Highly Restrictive Environments

Oleg Chunikhin | CTO, Kublr
Introductions

Oleg Chunikhin
CTO, Kublr

✓ 20 years in software architecture & development
✓ Working w/ Kubernetes since its release in 2015
✓ **Software architect behind Kublr**—an enterprise ready container management platform
✓ Twitter @olgch; @kublr

Like what you hear? Tweet at us!
Creating a Production-Grade Kubernetes Cluster

1. Install with kubeadm/other tools[^1,^2]
2. ...installer works its magic...
3. Done?

[^1]: https://kubernetes.io/docs/setup/independent/install-kubeadm/
[^2]: https://kubernetes.io/docs/setup/

@olgch; @kublr
Creating a Production-Grade Kubernetes Cluster

Unfortunately, it’s not that easy!

@olgch; @kublr
What We’ll Discuss Today

1. Cloud native, Kubernetes, and Enterprise
2. Enterprise Restrictions and Requirements
3. Kubernetes enterprise deployment patterns
4. Kubernetes solution categories and their limitations
5. On-premises struggles
Cloud Native

✓ Cloud Native Precursors
  ✓ SRE, DevOps, 12factor app
  ✓ API (management), Microservices
  ✓ Containers, Cloud, Virtualization

✓ Empower IT teams to respond to business requirements quickly, reliably, and predictably

✓ Larger Enterprises can benefit most, but adoption is lagging behind

@olgch; @kublr
Cloud Native Attributes

- Lightweight containers
- Language agnostic
- Microservices
- API
- Stateless/stateful separation
- Self-service infrastructure
- Isolated from OS/server deps
- Agile DevOps processes
- Highly automated
- Declarative resource mgmt

@olgch; @kublr
Enterprise Requirements

✓ Multiple/complex environments (On-prem, Clouds, Hybrid)
✓ Centralized management and governance
  Provisioning, Monitoring, Log Collection, IdM/AAA, Cost
✓ Integration with existing tools
✓ Security (Infrastructure, OS, IdM/AAA)
✓ Software management (Patches, Packages, Images)
Enterprise Constraints

✓ Separation of Responsibilities
  Infrastructure, Operations, Security, Legal

✓ Network Access (white/black-listing, air gap)

✓ Security Tools and Processes (infra, OS, platform, apps)

✓ OS, Platform, and Software Practices and Standards
  Vendor and version certification; configuration practices; custom package repositories; etc
Cloud Native Enterprise Requirements and Patterns

@olgch; @kublr
Cross-Team Responsibilities

✓ Large organizations often separate teams by:
  • Compute
  • Network
  • Traffic ingestion
  • Storage
  • Security

✓ “Cloud native” paradigm shift is necessary
Centralized Management

- Unification, standardization, governance
  - Centralized vs distributed management
- Management API
- RBAC and IdM/AAA; integration

@olgch; @kublr
Logging and Monitoring

✓ Centralized collection and analysis
✓ Integration with existing solutions
✓ RBAC for logs and metrics across teams
  • per project
  • per team
  • per environment

@olgch; @kublr
Security

- Identity Broker
- Fine-grained role-based access control (RBAC)
- IdM/AAA
- Secret management and support for external secret storage

- Cluster secrets storage/rotation
- Internal CA
- Support for external CA
- Infrastructure mgmt integration

@olgch; @kublr
K8S Security Tools and Best Practices

- Utilize RBAC
- SELinux/seccomp
- PodSecurityPolicies
- NetworkPolicy
- Authentication and Authorization Integration
  - OIDC, Web Hooks, Authenticating Proxy
- Admission Web Hooks

@olgch; @kublr
Audit

- Kubernetes API server audit
- Audit support for the logging and monitoring dashboards
- Audit support in the cluster provisioning tool (cluster install, update, upgrade, delete)
Complex Environment
Heterogeneous/Hybrid/On prem

✓ Infrastructure management differences
✓ Infrastructure automation
✓ Network connectivity and protection
Complex Environment
Isolated/Air Gap

- Where to get the required OS packages?
- How to provide the required container images?
- Binary repository (for helm and agents)?

@olgch; @kublr
Requirements | Support Existing Tooling

- Integration with existing processes and tools for deployment, logging and monitoring, security, software management etc.
Requirements | Cloud Native Platform

- Kubernetes
- Cloud native storage
- Cloud native DB
- Network policy

- Image management
- Backup and DR
- Integrated CI/CD

@olgch; @kublr
On Premises Struggles

- Pure bare metal limitations
- vSphere API interactions
- Realizing HA for Kubernetes
- Disaster recovery
- OS upgrades
- Security updates
- Kubernetes upgrades
- Air-gap/ offline mode

@olgch; @kublr
What are Your Options?

- Cloud provider managed Kubernetes
- Home grown solution
- 3rd party vendors
Cloud Provider Managed Solution

- Quick, easy, integrated, managed
- May not meet your requirements and/or regulations
- Access to masters and Kubernetes components in general
- No or limited K8S configuration customizations
- Support for on-prem / hybrid installations

@olgch; @kublr
Home Grown Solution

- Will cover your needs
- Requires extra time and efforts that could be spent on innovation
- With 4 major releases per year, it may be hard to keep up with upstream Kubernetes

@olgch; @kublr
Vendor Solution

✓ Will cover your needs
but
✓ Careful requirement definition and feature analysis is necessary; choose wisely!
✓ Custom development and integration may still be required

@olgch; @kublr
What’s Next?

- Infrastructure as a code
- Immutable Infrastructure
- CI/CD for infrastructure
- GitOps

@olgch; @kublr
Take Kublr for a test drive!
kublr.com/deploy

Free non-production license

@olgch; @kublr
Stay in touch! Signup for our newsletter at kublr.com

Oleg Chunikhin
CTO, Kublr
oleg@kublr.com
@olgch

Kublr | kublr.com
@kublr