

Guide BCP + Kubernetes PME

Resilience Cloud pour PME canadiennes

RTO/RPO, Velero, ArgoCD, Cloudflare Failover — cas BOTUM

Velero + **ArgoCD** + **Cloudflare**

Mars 2026 — Série Cloud Journey B18 (Final)

Introduction : la question que personne ne pose

Voici la question que personne ne pose lors de la migration cloud : Si notre infrastructure cloud tombe complètement ce soir, dans combien d'heures nos clients peuvent-ils reprendre leur travail ? Ce guide final de la serie Cloud Journey repond a cette question avec des strategies concretes, testees en production.

RTO et RPO : les deux metriques fondamentales

RTO (Recovery Time Objective) : le temps maximum acceptable entre la panne et la reprise.

RPO (Recovery Point Objective) : la quantite maximale de donnees perdables, exprimee en temps. Ces deux metriques definissent votre architecture DR et son cout.

Type d'application	RTO cible	RPO cible	Strategie recommandee
E-commerce / SaaS B2C	15-30 min	5 min	Active-Active
SaaS B2B / Apps metier	1-4 heures	15-30 min	Warm Standby
Outils internes / Back-office	4-24h	1-4h	Pilot Light
Services finances / Sante	Reglemente	Reglemente	OSFI E-21 / PHIPA

Strategies DR Cloud : comparatif cout vs RTO

Strategie	RTO	RPO	Cout mensuel	Pour qui
Cold Backup / Snapshots	4-24 heures	1-24h	5-10% infra	Outils internes, archives
Pilot Light	1-4 heures	15-60min	5-15% infra	Startups, SaaS B2B
Warm Standby	15-60 min	5-15min	20-40% infra	PME 50-500 emp., SLAs
Active-Active	< 5 min	< 1 min	80-100% infra	FinTech, eCommerce

Kubernetes pour la resilience

Auto-healing

Si un Pod crashe, K8s le redemarre. Si un Node tombe, K8s reschedule tous ses Pods. Les pannes de niveau 'serveur qui plante' sont gereses sans intervention humaine, en < 2 min.

Rolling deployments

Les mises a jour se deploient progressivement sans downtime. Si le nouveau Pod echoue son health check, le rollout s'arrete automatiquement.

Health checks

livenessProbe : detecte si l'app est en vie (sinon → redemarrage). readinessProbe : determine si le Pod est pret (sinon → retire du LB).

Resource limits

Kubernetes garantit a chaque Pod un minimum de CPU/RAM (requests) et l'empêche de consommer au-delà d'un maximum (limits). Isolation via cgroups.

Architecture haute disponibilité

Multi-AZ

Deployer vos workers K8s sur min 2 zones. Configuration : NodeAffinity + PodAntiAffinity. Cout additionnel : quasi nul.

Load balancing

AWS : ALB + AWS Load Balancer Controller. Azure : Application Gateway. GCP : Cloud Load Balancing. Ingress interne : nginx-ingress ou Traefik.

Circuit breaker

Protège contre les pannes en cascade. Si un service dépendant ralentit, le circuit breaker 'ouvre' la connexion. Implementations : Istio, Envoy, Resilience4j.

Tests de resilience : chaos engineering et fire drills

Un BCP non testé est une fiction. La vraie question : avons-nous prouvé que ce plan fonctionne ?

- LitmusChaos : outil open-source pour K8s — simule pannes de Pods, Nodes, latences réseau
- Fire drills trimestriels : simuler un disaster complet, chronométrer le temps de reprise réel
- Runbooks : listes de commandes copiables, stockées dans git, testées à chaque fire drill
- Netflix Simian Army (référence) : Chaos Monkey, Latency Monkey, Conformity Monkey

Conformité canadienne : OSFI E-21, PHIPA, LPRPDE

OSFI E-21

Exigences explicites : définition RTO/RPO, tests réguliers de reprise, documentation des dépendances critiques. Concerne les institutions financières fédérales et les fintechs partenaires.

PHIPA (Ontario)

Les dépositaires d'informations de santé doivent maintenir des plans de continuité pour assurer la disponibilité des données de santé. Une perte de données non couverte par un BCP peut constituer une violation.

LPRPDE / Loi 25

Obligation de notification en cas d'atteinte aux mesures de sécurité. Un BCP documenté et testé démontre la diligence raisonnable requise.

Stack recommandée PME

Velero

Backup K8s : ressources + volumes persistants (PVCs). Config type : backup/heure, rétention 7j, snapshot/jour, rétention 30j. Destinations : S3, Azure Blob, GCS.

ArgoCD / Flux (GitOps DR)

Cluster DR synchronise depuis git. En cas de disaster : 1) Bascule DNS. 2) ArgoCD a deja deploye toutes vos apps. 3) Velero restore les donnees.

Cloudflare Load Balancing

Health checks → failover DNS automatique vers cluster DR. TTL 30s. Plan Business (\$200/mois) suffit pour la plupart des PME. Alternative : Route 53 Health Checks.

Prometheus + Alertmanager

Definir des SLOs. Alerter a 99.9% avant la panne a 99.5%. PagerDuty ou OpsGenie pour l'astreinte.

Cas concret BOTUM : RTO atteint en 23 minutes

Novembre 2024 : corruption etcd suite a une mise a jour de noeud mal orchestree. 14h37 : premiers 503. 14h41 : alerte Alertmanager. Voici la timeline complete :

Temps	Evenement	Action
T+0 min	Detection automatique (etcd unhealthy)	Alertmanager → PagerDuty
T+4 min	Confirmation corruption etcd via kubectl	Ingenieur d'astreinte
T+8 min	Decision basculement vers cluster DR	War room (Slack)
T+10 min	Cloudflare DNS failover declenche (TTL 30s)	Automatique
T+12 min	ArgoCD DR confirme apps deployees + saines	Automatique (GitOps)
T+18 min	Velero restore dernieres donnees	Script runbook
T+23 min	100% trafic sur cluster DR — services OK	Validation manuelle

Ce qui nous a sauves : cluster DR synchronise via ArgoCD (meme repo git), snapshot Velero 8 min avant l'incident, runbook teste lors du fire drill de septembre. Sans le BCP : estimation 4-6h de downtime complet.

Conclusion : fin de serie, debut de la resilience

18 billets, un parcours cloud complet. La resilience n'est pas un etat que vous atteignez — c'est une pratique que vous exercez. Commencez la ou vous etes : installez Velero cette semaine, ecrivez un runbook, planifiez un fire drill. Etape par etape.

Construire votre BCP cloud avec BOTUM

Plan de continuite, architecture resiliente, Kubernetes manage — les equipes BOTUM vous accompagnent de la conception a l'implementation.

→ www.botum.ca/contact