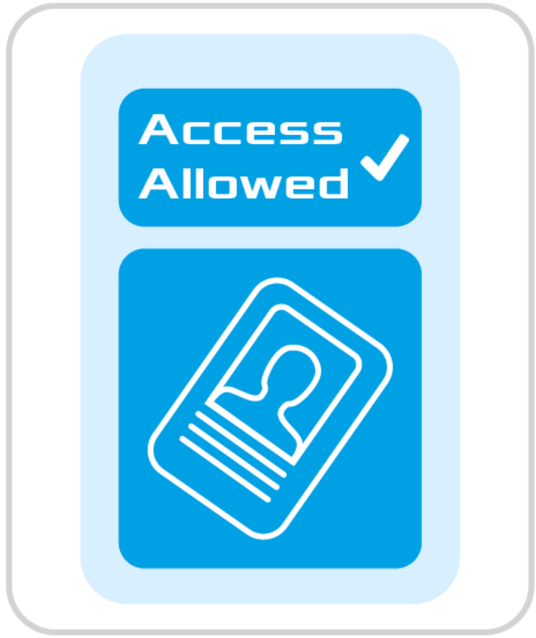
СЛЁРМ
+
Southbridge

Авторизация в кластере

Сергей Бондарев
Архитектор Southbridge

RBAC

- Role
- RoleBinding
- ClusterRole
- ClusterRoleBinding
- ServiceAccount



Role

```
# GET /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name}
- apiGroups: ["extensions", "networking.k8s.io"]
   resources: ["ingresses"]
   verbs: ["get", "list", "watch", "create", "update", "patch", "delete"]

# GET /api/v1/namespaces/{namespace}/pods/{name}/log
- apiGroups: [""]                    # "" indicates the core API group
  resources: ["pods", "pods/log"]
  verbs: ["get", "list"]
```

https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.18

https://kubernetes.io/docs/reference/access-authn-authz/authorization/

RoleBinding

```
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: nginx-ingress
subjects:
- kind: ServiceAccount
  name: nginx-ingress
  namespace: ingress-nginx
- kind: User
  name: jane                # "name" is case sensitive
  apiGroup: rbac.authorization.k8s.io
- kind: Group
  name: developer       # for example organization in user certificate
  apiGroup: rbac.authorization.k8s.io
```

Kubectl config

```
kubectl config set-cluster slurm.io \
  --server https://172.20.100.2:6443 \
  --certificate-authority=/etc/kubernetes/pki/ca.crt \
  --embed-certs=true

kubectl config set-credentials username \
  --token BFG9000js23..==

kubectl config set-context slurm.io \
  --user username \
  --cluster slurm.io \
  --namespace default

kubectl config use-context slurm.io
```

Resource Quota

Устанавливает количество доступных ресурсов
и объектов для нэймспэйса в кластере

Реквесты

Лимиты

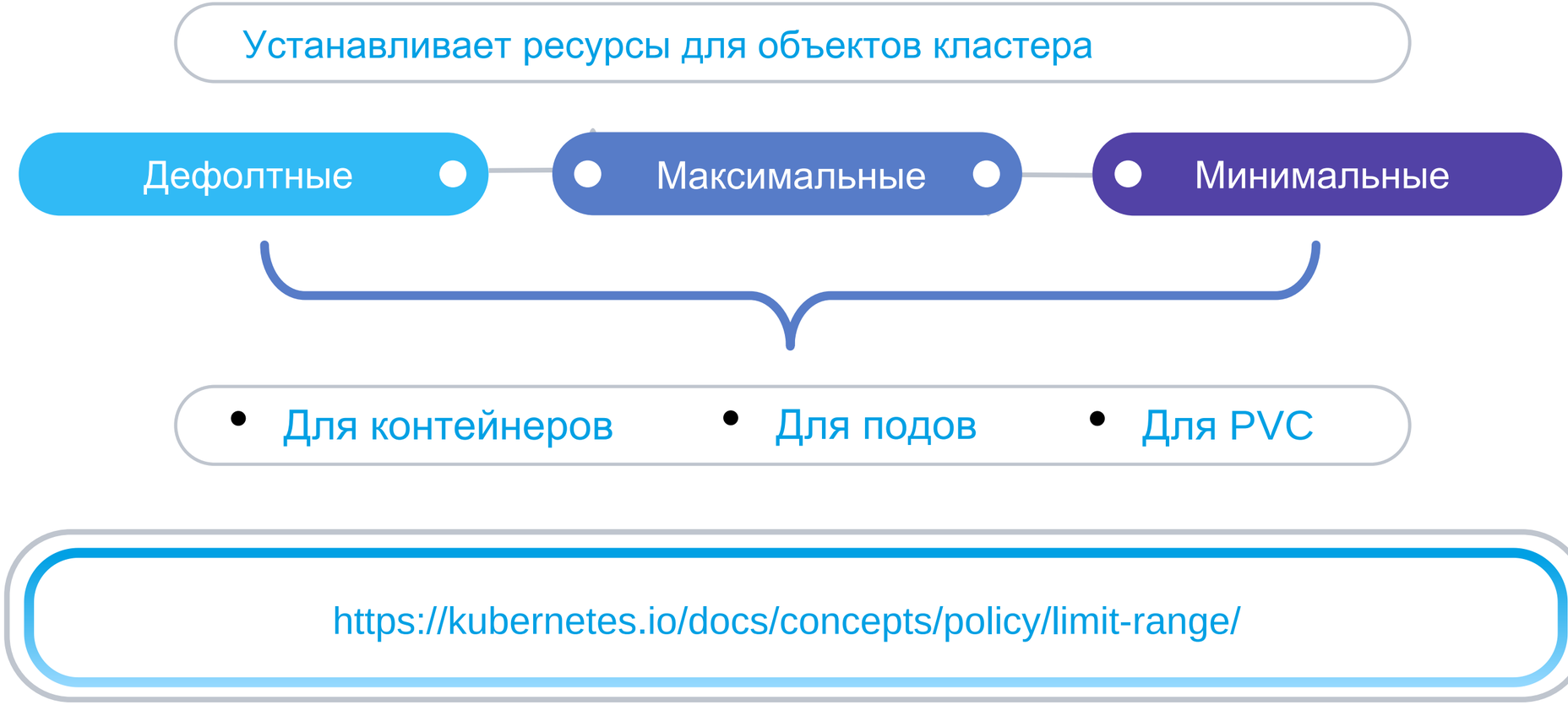
Сервисы

Поды

…

https://kubernetes.io/docs/concepts/policy/resource-quotas/

Limit Ranges



Pod Security Policy

- Контролирует аспекты безопасности в описании Pod’ов
- Включается как admission controller plugin “PodSecurityPolicy”
  - При включении запрещает запуск Podов без PSP

https://kubernetes.io/docs/concepts/policy/pod-security-policy/