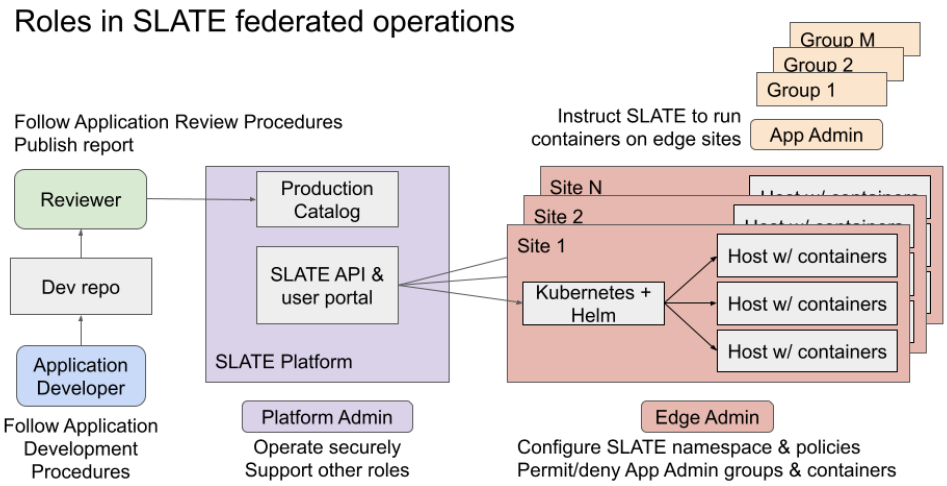
# SLATE Security Summary

| Version | Comment | Effective date |
|---|---|---|
| 1.0 | Initial version | April 15, 2022 |

## Overview

The SLATE system implements a distributed federated operations model[1]. An *Edge Site* (a resource providing organization) can (but is not required to) delegate service operations (including but not limited to initial application deployment and configuration, updates and monitoring) to *Application Administrators*[2], members of a privileged *Group* selected by the Edge Site. A *Cluster* presents edge resources using the Kubernetes API into a restricted namespace. The Edge Cluster itself requires privileged Kubernetes system administration (Kubernetes release updates, certificates, etc.) and is the responsibility of Edge Administrators, i.e. staff at the Edge Site. Application Administrators use the SLATE API service (not the Kubernetes API directly), which is hosted by the University of Chicago and maintained by SLATE *Platform Administrators*, to manage permitted application containers on the Edge Cluster. *Application Developers* create and maintain software images and deployment charts (recipes) according to documented procedures for publishing into a production catalog. *Application Reviewers* ensure proper procedures have been followed which includes standard information regarding source origin, configuration and operation details, and potentially other metadata. The actors and roles are summarized in the figure below.




[1] https://inspirehep.net/literature/1832212
[2] SLATE Roles: https://slateci.io/docs/concepts/index.html

*This document is a policy of the SLATE (Services Layer at the Edge) project, supported by the National Science Foundation Office of Advanced Cyberinfrastructure: "CIF21 DIBBs: EI: SLATE and the Mobility of Capability", award number OAC-1724821.*

Edge Administrators, Application Administrators, and Platform Administrators agree to the SLATE Acceptable Use Policy[3] (AUP) before receiving a token to access the SLATE API. Each role has a documented set of obligations[4,5,6,7]. Platform Administrators follow a documented set of security policies and procedures[8]. These policies and procedures are in accordance with *Security for Collaborating Infrastructures Trust Framework, v2* [9].

## Container Security

Documented development and review procedures[10] guide Application Developers and Application Reviewers to ensure that pertinent configuration, operational, and security characteristics of the application are addressed in a standard README file format. See further https://github.com/slateci/slate-catalog.

These procedures do not enforce a predetermined set of security controls. They do inform security and other operational considerations that Edge Administrators take into account before authorizing an application from the Production Catalog to run at their site. The CIS Docker[11] and CIS Kubernetes Benchmarks[12] are suitably reflected in the Application Development and Review Procedures in addition to other obligations.

## SLATE Platform security

Comprehensive security policies and procedures were produced through an engagement with TrustedCI and an extended engagement with the University of Chicago Information Security team. These policies and procedures address management of credentials and access to each component system of the SLATE Platform, technical (and physical, where appropriate) security controls pertaining to those component systems, incident response and disaster recovery procedures, privacy policy, and change management.

An Overview of SLATE Platform Internals and Security describing its various components, their relationships, data flows, and major security mechanisms, is available[13].

---

[3] SLATE AUP: https://slateci.io/docs/security-and-policies/acceptable-use.html
[4] SLATE Edge Administrator Obligations: https://slateci.io/docs/security-and-policies/edge-administrator-obligations.html
[5] SLATE Application Administrator Obligations:
[6] SLATE Application Developer Obligations: https://slateci.io/docs/security-and-policies/slate-application-developer-obligations.html
[7] SLATE Application Reviewer Obligations: https://slateci.io/docs/security-and-policies/slate-application-reviewer-obligations.html
[8] SLATE Security and Policies, https://slateci.io/docs/security-and-policies/index.html
[9] Security for Collaborating Infrastructures, version 2, https://wise-community.org/sci/
[10] SLATE Application Development and Review Procedures: https://slateci.io/docs/security-and-policies/slate-application-developer-and-review-procedures.html
[11] CIS - Securing Docker: https://www.cisecurity.org/benchmark/docker/
[12] CIS - Securing Kubernetes: https://www.cisecurity.org/benchmark/kubernetes/
[13] Overview of SLATE Platform Internals and Security, https://slateci.io/docs/security-and-policies/overview-of-slate-platform-internals-and-security.html

*This document is a policy of the SLATE (Services Layer at the Edge) project, supported by the National Science Foundation Office of Advanced Cyberinfrastructure: "CIF21 DIBBs: EI: SLATE and the Mobility of Capability", award number OAC-1724821.*

# Network access to Edge Sites

## Connections in-bound to an Edge Site

### SLATE Control channel

In order for an Edge Cluster to federate with the SLATE Platform, the cluster's Kubernetes master's API port must be able to listen for requests from the SLATE Platform. This is port 6443/TCP on the Kubernetes master node, and the traffic for which it listens is HTTPS, specifically the Kubernetes API. The only host from which these requests should be allowed to originate is api.slateci.io (128.135.158.222). The security of these communications is important for ensuring that the SLATE Edge Cluster is only accessed in a properly authenticated manner. Configuration of this security may be managed by the administrator of the Kubernetes cluster at cluster configuration time. See the kube-apiserver documentation[14] for details, particularly the --tls-cipher-suites and --tls-min-version options.

## Connections out-bound from an Edge Site

### SLATE request fulfillment

The SLATE Edge Cluster's Kubernetes master node may need to pull information from several sources to fulfill a request received from the SLATE API server. When an Application Administrator asks the API server to run an application on an Edge Cluster, the API server fetches the application's Helm chart from the Production Catalog and sends it to the Edge Cluster's Kubernetes master node for implementation. This may require it to fetch a container image from one of three repositories: DockerHub, the Open Science Grid's Harbor service, or SOTERIA.

### User access to the SLATE User Portal

Edge Administrators instruct SLATE about which SLATE users may act as Application Administrators on their Edge Cluster and which applications in the Production Catalog are permitted (or denied) to run on their Edge Cluster. Application Administrators who have been authorized by an Edge Administrator may instruct SLATE to run an application on the Edge Cluster that has been permitted by the Edge Administrator. These instructions are issued by these SLATE users either by accessing the SLATE User Portal with a web browser or by using a SLATE CLI tool available for this purpose. Both establish outbound connections to the SLATE User Portal on port 443/TCP. Upon authenticating users and their authority to issue such instructions, the SLATE User Portal uses the SLATE API to issue corresponding requests to the Edge Cluster as may be needed to carry out the instructions.

[14] kube-apiserver documentation: https://kubernetes.io/docs/reference/command-line-tools-reference/kube-apiserver/

### Application connection requirements

Each application authorized to be run on an Edge Cluster has its own configuration requirements, including network ports that may need to be open. These details are found in the application's README[15] file in the Production Catalog.

## Tabular summary of in-bound connections to and out-bound connections from an Edge Cluster

### **In-bound** Edge Cluster connections (excluding application-specific connections)

| **Port** | **Client** | **Server** | **Purpose** |
|---|---|---|---|
| 6443/TCP | api.slateci.io | SLATE Edge Cluster's Kubernetes master node | SLATE control channel |

### **Out-bound** Edge Cluster connections (excluding application-specific connections)

| **Port** | **Client(s)** | **Server(s)** | **Purpose** |
|---|---|---|---|
| 443/TCP | SLATE Edge Cluster's Kubernetes master node | https://hub.docker.com<br>https://hub.opensciencegrid.org | Fetch container images of applications in the Production Catalog |
| 443/TCP | SLATE user's End User Device | https://portal.slateci.io | SLATE user instructions to the SLATE Platform |

[15] Template for application README - see Appendix 1 of SLATE Application Developer and Review Procedures:
https://slateci.io/docs/security-and-policies/slate-application-developer-and-review-procedures.html

*This document is a policy of the SLATE (Services Layer at the Edge) project, supported by the National Science Foundation Office of Advanced Cyberinfrastructure: "CIF21 DIBBs: EI: SLATE and the Mobility of Capability", award number OAC-1724821.*