SLATE Status for WISE

Chris Weaver for the SLATE Team
WISE @ NSF Cybersecurity Summit 2019

SLATE Workflows

Workflows

- Edge Providers Add CI to SLATE
- SLATE Application Developers Submit Applications to SLATE
- SLATE Group Requests Application Service to Execute
- State Preservation Via Cloud
- Group Creation and Management

Edge Providers Add CI to SLATE

A SLATE Edge Provider desiring to add their Kubernetes cluster to SLATE submits a request through the API Server. This process

- installs the SLATE software on the Kubernetes cluster
- generates certificates for SLATE on the cluster
- registers the SLATE Edge Provider to SLATE

During the install, the SLATE client is pulled from SLATE build artifact server. At any point after registration, the Edge Providers' SLATE Edge Administrator can *whitelist* what SLATE Groups are allowed to execute on their CI.

Edge Providers Add CI to SLATE

Observations:

- Communication between the SLATE Edge Provider and the SLATE API server is done through TLS
- There is no checksum done on the SLATE client
- Kubernetes clusters must have inbound and outbound access to the API Server
- SLATE client relies on 3rd party NRP-module
  - A copy of the NRP module source with SLATE Core
- Currently anyone with a Globus account can register and add a cluster to SLATE

Risks:

- For CI owners, who are trusting that their CI is used appropriately & securely:
  - DoS, AUP violation, unauthorized access, network bandwidth (perhaps related to DoS), reputation, contention of resources (not really concerned with side-channel issues),

Planned Actions:

- Provide checksums for client binaries

SLATE Application Developers Submit Applications

- SLATE Application Developers desiring to add their containers to SLATE first must produce a Helm chart.
- They then submit their Helm chart to GitHub as a Pull Request. This triggers an automated continuous integration process running on the Jenkins VM that vets the Helm chart for correct syntax.
- Once the Helm chart is in the repo, a SLATE Core Developer is notified, and can begin the process of manually vetting the container(s) the Helm chart specifices.
- If satisfied, the Helm chart is moved to the production category, at which point the continuous integration tool publishes the Helm chart and containers.

SLATE Application Developers Submit Applications

Observations:

- Jenkins VM
  - Open to Internet on both ssh and https
  - AuthN done by SSH Public-key for SLATE Admins
  - No hIDS or nIDS protecting server, no vulnerability scans, no automated backups
  - Logs are forwarded to a central logging server at U. Chicago
- Clair scanner checks syntax on Helm charts, but it is currently disabled (too noisy and slow)
  - SLATE Core is interested at looking at Trivy as a potential replacement
- GitHub does not currently require MFA

Risks:

- To Application Developer:
  - Reputation (based on code being modified after upload, i.e., integrity),
- To SLATE
  - Buggy/malware code being submitted
  - Reputation

Planned Actions:

- Investigate scanners with TrustedCI
- Configure backups of Jenkins VM to U. Chicago backup server

SLATE Group Requests Application Service to Execute

- When a SLATE Group desires to execute a Service Application, they connect to the API Server request execution of said service.
- If the SLATE Group is whitelisted at the SLATE Edge Provider specified in the request, then the API server instantiates the service on the Edge Provider's Kubernetes cluster.
- SLATE Group users can check on the status of their requests via the API Server.

SLATE Group Requests Application Service to Execute

Observations:

- The 'grant' is done by verifying that the Application Service group is whitelisted by the SLATE Edge Administrator
- SLATE Group users can view catalog and list of SLATE Edge Providers
- Kubernetes namespace & secrets are used to isolate groups
    - Any user in a group can see any other user in that group’s data
- When a user account is created, they get a token, which has the same permissions as the user - there is currently no option for a read-only token

Risks:

- For Application User:
    - Loss of science due to tampered/buggy software
    - Reputation (if application misbehaves)
    - Confidentiality (of embargoed data)
- For CI Provider:
    - DoS
    - Reputation
- For SLATE:
    - Reputation
    - Lost Functionality
    - Lost time of work troubleshooting

State Preservation Via Cloud

- SLATE Admins preserve the **state** of SLATE users, groups, and application instances by leveraging DynamoDB service within AWS.
- All Service Provider information and Application User requests are stored within the database.
- Sensitive data (credentials) submitted by users is stored encrypted
- The Helm charts of the Application Catalog are hosted on the SLATE Jenkins server
- The Docker images of the Application Catalog are hosted on Docker Hub, in the slateci organization

State Preservation Via Cloud

Observations:

- Connections to DynamoDB are encrypted
- Only the SLATE API server should be connecting, but they currently don’t have any IP restrictions because during development debugging has been necessary, and Dynamo is shared with other services

Risk:

- For SLATE:
    - Sensitive credentials
- Application Developers:
    - Sensitive credentials
        - Could allow unauthorized access to science

Group Creation and Management

- Any SLATE user can create a Group (or multiple groups).
- All members within a group have equivalent privileges to administer the Group and its resources, which include any Edge Clusters it has joined to the Federation and any Application Instances it has launched.
- Any member of a Group may add Users to or remove users from the Group.

Status of Security Policies

In Progress and Planned Policies

In Progress	Planned
Acceptable Use Policy Asset Management Policy (part of MISPP) Incident Response Procedures	Disaster Recovery Policy Access Control Policy Edge Admin/Federation Policy Information Classification Policy (MISPP) Personnel Exit Checklist Mobile Compute Policy Network Security Policy Password Policy Physical/Environmental Security Policy Privacy Policy Remote Access Policy (MISPP)

Not planned as separate policy: Training and Awareness Policy

Asset Management Policy

- Maintain a spreadsheet of Information Assets
  - Credentials, non-credential data, and information systems
- Record where data is stored and to whom data and systems are accessible
- Note possible impacts of loss of availability, confidentiality, and integrity, as applicable

Incident Response Policy

- Define a group of team members who are responsible for carrying out documented procedures, and reporting to the Information Security Officer when appropriate
- Much of focus is on how and when information sharing is needed, since there are many stakeholders
  - For example, if a catalog application is determined to have a security vulnerability, it may be necessary to notify:
    - The developer or maintainer of the application
    - Users who were running instances of the application
    - Sites on whose infrastructure instances of the application were running

Incident Response Policy — Types of Incidents

- User credential compromise
- Edge cluster credential compromise
- Vulnerable application
- Exploited application
- Container image repository compromise
- Build server compromise
- Source code repository compromise