About Process Server Load Balancing and Clustering on Secure Agent

Technical Note

Dated: May 2017
Contents

Overview .............................................................................................................................................. 3
Choosing What’s Best for your Needs .................................................................................................. 3
Details ................................................................................................................................................ 5
  HTTP/HTTPS Listener support ....................................................................................................... 5
  Configuring a Cluster ...................................................................................................................... 5
  Selecting the Master Node’s Database .......................................................................................... 6
    Cluster failure scenarios .............................................................................................................. 7
    Cluster upgrades ......................................................................................................................... 7
  Load Balancing .............................................................................................................................. 8
Overview

The May 2017 release of the ICRT service will deliver the ability to load balance Cloud-to-Agent requests using the Agent Group capability and optionally deliver a high-availability deployment configuration.

When you create a Secure Agent, it is added to its own group by default. If you have the Secure Agent Cluster license, you can add multiple agents to one Secure Agent group. All agents within a group must be co-located. You can also mix Linux and Microsoft Windows agents within a group. The Cloud Application Integration assets you deploy to a Secure Agent group are processes, connections, or service connectors.

Agent Groups can be used to deploy Process Server services on premises in two configurations:

- **Load Balanced Process Server Configuration** to balance the workload across Process Server instances within a group. Add multiple agents to a group to balance the distribution of tasks across Process Servers. When the runtime environment is a Secure Agent group with multiple agents, the group dispatches incoming requests to available agents in a round-robin fashion.

- **Clustered Process Server Configuration** to provide high availability with Process Server instance clustering. You can cluster two or more Agent Group members into a single logical Process Server instance. When you cluster Process Server instances, if a node failure occurs, running process instances failover kicks in to recover the instance on another node. The Clustered Process Server Configuration inherits the capabilities of the Load Balanced configuration in that calls from Cloud to Process Server of a Secure Agent Group are load balanced in a round-robin fashion.

Choosing What’s Best for your Needs

Most customers will only need to use the load balancing capabilities of an Agent Group. Stateless calls (e.g. OData requests or stateless short-lived processes) typically only need to be load balanced between members of the group. If all you are doing is processing stateless requests the Agent Group Load Balancing capability is sufficient. If you are processing stateless requests or utilizing the agent only to serve OData requests, we recommend that you do not use a Clustered configuration.

Consider clustering if you are processing long-running processes and need to perform recovery of payload processing from one node to another transparently. A Clustered configuration requires that a single database server be shared with all cluster nodes. A Load Balanced configuration on the other hand distributes writes across each Process Server’s respective database. In some cases, the Load Balanced configuration will perform better than a clustered configuration because writing of process execution state information to the database is distributed across several databases.

If you cluster Process Servers, you must select a Master node. If you are not clustering Process Server nodes then each agent will utilize its own PostgreSQL database. When you employ clustering, the Secure Agent group, and all the Secure Agents within the group, share the PostgreSQL database of the Master node. Do not add more than one Master node to a Secure Agent group.

The following image depicts the cluster configuration of a Process Server cluster with four Secure Agents in an Agent Group sharing a single database server instance:
You can deploy process, connector and connection definitions to a Secure Agent group or to a specific Secure Agent within a group. Informatica Cloud deploys definitions to Process Server instances in the following manner:

- **Load Balanced Configuration.** You can deploy definitions to Process Server instances of a Secure Agent group in a Load Balanced configuration. Here, all Process Server instances within that group will be informed of the existence of the new or update definition. In that configuration, the deployment of definitions is automatically performed to every node of the group for you.

- **Clustered Configuration.** You can deploy definitions to Process Server instances of a Secure Agent group in a Clustered configuration. Here, all Process Server instances within that group will be informed of the existence of the new or update definition. In that configuration, the deployment of definitions is performed to the Master node of the group which informs other nodes of updates.

You can also deploy a definition to a specific Process Server instance of a Secure Agent of a Secure Agent group. Here, the definition is only published to the specific node. Other Process Server instances of the group will not be informed and would not be able to process requests making use of that definition. Though possible, this is an unlikely use of the capability.
Details
The platform improvements made to the Secure Agent released broadly with the Spring 2017 enabled the following new capabilities:

- The Process Server service is a new engine of the Secure Agent platform that executes independently of other engines to help overall resilience.
- The underlying Secure Agent database for Secure Agent version 33.0 is PostgreSQL. The PostgreSQL database can handle a larger amount of data than the earlier underlying database, H2.
- You can instantiate processes published to a Secure Agent from within the enterprise through HTTPS/REST or HTTPS/SOAP calls in addition to JMS and AMQP queues and topics.

HTTP/HTTPS Listener support
HTTP/HTTPS listeners are now available on ports 7080/7443 respectively and be re-configured to use other ports within the ICS console.

Configuring a Cluster
Should this be desired, you can cluster Process Server services to ensure high-availability of the execution of process instances within a cluster. The Process Server packaged and distributed to the agent has the same capabilities as the one executing in the Cloud. The way it operates is dependent on how the server is configured. For example, the Process Server instance within an agent executes in a single-tenant configuration.

The configuration is managed transparently for you when the agent is declared as “cluster-enabled”. When a Process Server is cluster-enabled, configuration changes enable this mode of operation.
In addition to this, a common database needs to be configured for shared use by Process Server instances. By being cluster-enabled, you can execute processes on any node of the cluster. If a failure occurs, where one of the nodes fails and no longer available, another member of the cluster automatically recovers and continues any process instances which were executing at the point of failure.

An example of the cluster configuration is shown above. By setting the cluster-enabled flag to true you transform the Process Server service on the agent as a cluster-aware service. There are additional properties in the cluster section that are relevant when clustering is enabled. The name of the cluster can optionally be specified; however, the default should suffice for most installations. There is also a primary-node setting, that controls which node is the Master node. The Master node is responsible for hosting the PostgreSQL database, as well as managing database updates. If this flag is set to false on a cluster member, it will have the effect of not starting the PostgreSQL database for that node.

Finally, the load-balance-url is used to optionally specify the URL of a load balancer you deploy to load-balance requests to the agent from within your network. This controls the URL generation listed in Process Designer. If specified this value will be used as part of the URL generation. If left blank, then the default logic will be retained which uses the agent name as part of the URL reference. Summarizing, Cloud to Agent Group requests are load balanced by Informatica Cloud. The load balancer you configure is intended for you to load balance incoming request from your private network. For example, if you run agents within AWS, an AWS ELB load balancer can be used.

**Selecting the Master Node’s Database**

To configure the note to use the Master node’s database use the Process Server > db settings. The url specifies the connection URL of the agent hosting the common database. As a best practice, an agent which is not receiving service requests should be chosen to host the database. This will prevent performance issues by dedicating the agent for database transaction processing, and nothing else.
Cluster failure scenarios

A server hosting the agent application has the potential for shutting down without notice for any number of reasons. The reason may not be caused by the agent application itself but something out of its control. In the case of the agent shutting down which does not host the PostgreSQL database, it is expected that the other node in the cluster will detect this situation. It will then start the process of recovering any work which the failed node was processing and then execute those processes to completion. In the case where an agent hosting the database goes down, then all agents in the cluster will not be able to continue until it is resumed. The database is currently the single point of failure for a clustered environment. Options will be provided in the future to eliminate this single point of failure.

Unless there is a need to ensure process instance recovery, customers should consider the use of the Load Balancing configuration.

Cluster upgrades

Informatica Cloud distributes upgrades and patches periodically. For Process Server service upgrades, where there is only a minor revision change, the user will have the ability to upgrade their agent at their convenience. However, a change in major revision will trigger an upgrade of the process-engine package with which the Process Server service is distributed. During this upgrade, the PostgreSQL database will remain running unless that package itself has had a major revision change. In that case, the agent hosting the database will trigger a database restart to upgrade. Currently this should be the only source of downtime for a cluster. Future development efforts in this area need to account for higher resiliency of the database.
Load Balancing

The Agent Group concept provides the means to distribute Cloud-to-Agent requests in round-robin manner. If an agent is not available no further requests will be sent to the agent after several retry attempts.

To use this capability (as is the case for the Clustered Configuration) deploy processes to an Agent Group as depicted below. The "Run Process On" drop-down specifies where process instances, connectors and service connectors are deployed to and execute.

![Property Sheet](Image)

The drop-down list presents a list of all know agent groups in the Org as well as all agents known to the Org. When the user targets the "AvosCluster" agent group, all requests for execution of this process will be distributed among active members of that agent group.

Requests from the cloud are then load balanced by the agent group configuration, but when calling externally we may want to allow for load balancing control as well. A customer may choose to optionally configure a separate load balancer in front of their agents to use when invoking agent processes directly. If they choose to define such a configuration we would want to change the URL generation which we display to user in Process Designer so this is included as part of the URL. The Informatica Cloud properties for the Process Server service provide for this. These are controlled in the cluster → load-balance-url setting as shown below. If a value is set here, then we will use this in the URL generation we show to user and if nothing is set we default to URL generation which targets the specific agent. This configuration option is shown below.