

Technical Disclosure Commons

Defensive Publications Series

November 2023

On-demand Server-less Log Multiplexer for Single-Container Processes

Alankrit Kharbanda

Sowmya Dayanand

AJ Ortega

Follow this and additional works at: https://www.tdcommons.org/dpubs_series

Recommended Citation

Kharbanda, Alankrit; Dayanand, Sowmya; and Ortega, AJ, "On-demand Server-less Log Multiplexer for Single-Container Processes", Technical Disclosure Commons, (November 14, 2023)
https://www.tdcommons.org/dpubs_series/6416



This work is licensed under a [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/).

This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

On-demand Server-less Log Multiplexer for Single-Container Processes

ABSTRACT

In single-container processes, which limit the number of compute processes per container, the capture and transmission of logs can become infeasible because a separate, disk-based, log-capture and emission process is disallowed. This disclosure describes a log multiplexer that uses memory-multiplexing techniques to transport logs to multiple destinations within a single-container process. The log multiplexer interfaces with an in-memory, single-process, shim layer that encompasses a generic, single-container application, becoming a single-point pipe pass-through for all emitted logs. Based on a static or a dynamic configuration provided to the application, the single-point pass-through multiplexer determines the endpoints where the logs are to be redirected. Acting as a pass-through multiplexer for logs within single-process memory, the logger enables fast and dynamic log emission to multiple endpoints.

KEYWORDS

- Log capture
- Log multiplexer
- Data analytics
- Data processing
- Container
- Shim layer
- Log-file transport
- Single-container process
- Single-process deployment

BACKGROUND

Data analytics and processing applications in a private or distributed cloud deploy code that generates logs at several points. The log files are typically stored on the root system until moved to a more permanent destination. However, in single-container processes, which limit the number of compute processes per container, the capture and transmission of logs can become infeasible because a separate, disk-based, log-capture and emission process is disallowed.

Although log-capture mechanisms exist that read logs from disk and send them to multiple destinations, these do not operate from memory and are incompatible with single-process deployments.

DESCRIPTION

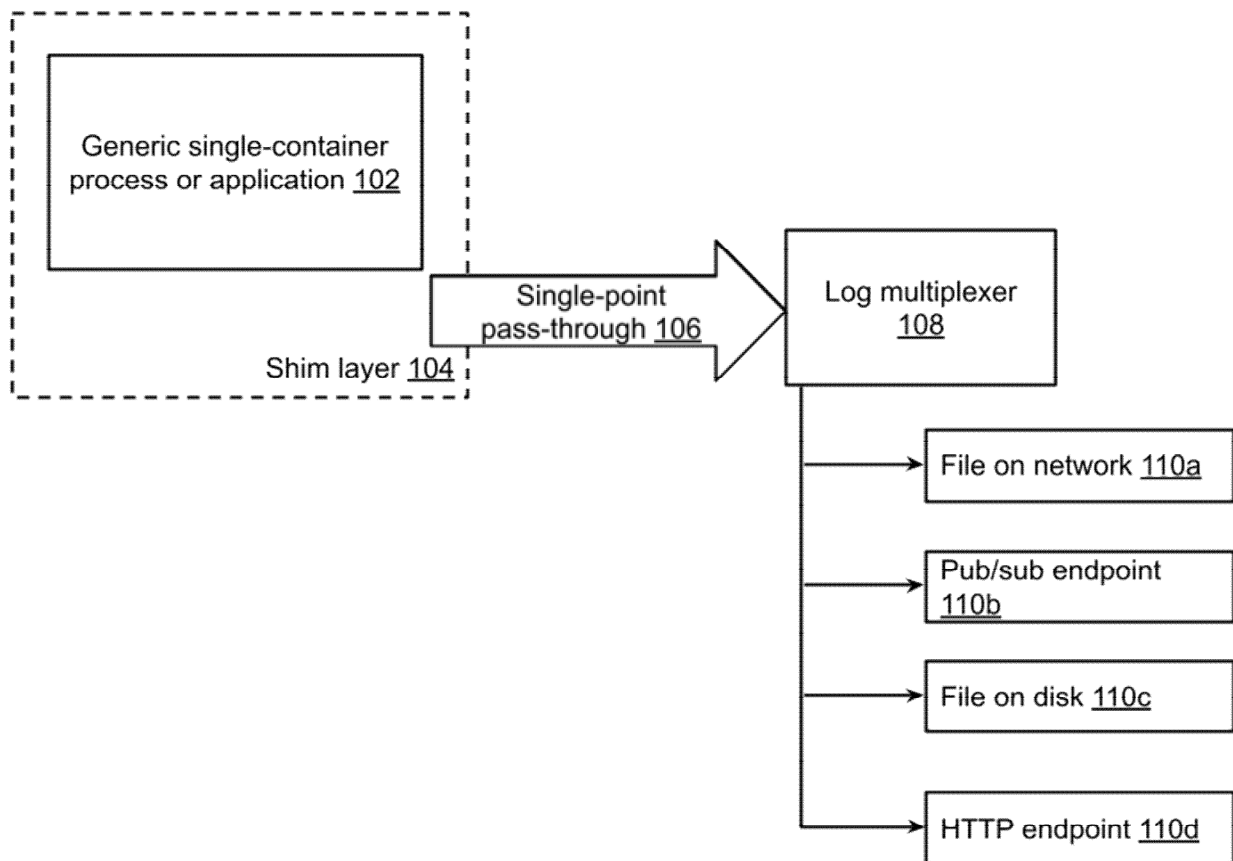


Fig. 1: On-demand server-less log multiplexer for single-container processes

This disclosure describes a log multiplexer that uses memory-multiplexing techniques to transport logs to multiple destinations within a single-container process. As illustrated in Fig. 1, the log multiplexer (108) interfaces with an in-memory, single-process shim layer (104) that encompasses a generic, single-container process or application (102), becoming a single-point pipe pass-through (106) for all emitted logs. Based on a static or a dynamic configuration provided to the application, the single-point pass-through multiplexer determines the endpoints where the log is to be redirected. Example endpoints include a file on network (110a), a pub-sub endpoint (110b), a file on disk (110c), an HTTP endpoint (110d), etc.

In contrast to a conventional application logger, the described shim-layer-based logger enables in-memory, configuration-based multiplexing of log destinations. The configuration can be static, e.g., provided at the start of application in a file or from a database, or dynamic, e.g., pulled by the application periodically during runtime from a database or an HTTP endpoint. Log emission to a given endpoint can be serial or parallel. Log-buffering and/or failure-retry mechanisms to a given endpoint can (or need not) be included in the logging procedures.

In contrast to currently available logging techniques that read logs out of disk, the described logger operates within memory. The described logger can therefore act as a pass-through multiplexer for logs within single-process memory, enabling more dynamic and faster log emission than existing techniques. The logger can also be a no-op, enabling it to ignore logs and to not write them to disk or to a remote endpoint. Such a no-op logger can act as a test-op that has logic to verify that correct logs are passed, a useful feature in testing environments. The described in-memory logger can be used as a remote deployed agent under single-process restrictions, e.g., in cloud deployments.

CONCLUSION

This disclosure describes a log multiplexer that uses memory-multiplexing techniques to transport logs to multiple destinations within a single-container process. The log multiplexer interfaces with an in-memory, single-process, shim layer that encompasses a generic, single-container application, becoming a single-point pipe pass-through for all emitted logs. Based on a static or a dynamic configuration provided to the application, the single-point pass-through multiplexer determines the endpoints where the logs are to be redirected. Acting as a pass-through multiplexer for logs within single-process memory, the logger enables fast and dynamic log emission to multiple endpoints.