March 2020

SECURE EXECUTION OF UNTRUSTED CODE

HP INC

Secure Execution of Untrusted Code

Abstract: Untrusted solutions embedded in an electronic device are executed securely and without compromising the security or the performance of the device with the use of a container technology.

This disclosure relates to the field of computer security.

A technique is disclosed that runs untrusted solutions embedded in an electronic device securely and without compromising the security or the performance of the device.

Many printers can run embedded solutions, which enables partners to build and deploy their own solutions on the device. However, partner applications are considered to be untrusted code, and therefore the printer manufacturer may need to perform a stringent validation and verification process (V&V) for ensuring the quality of the device. These partner solutions are run thru numerous manual checks, and then "cleansed" (to eliminate possible problems) before being deploying on the device.

According to the present disclosure, and as understood with reference to the Figure, logically isolated entities 10 are created using a container technology such as LXC. LXC containers are essentially operating system containers that enable process isolation while running on the same device. LXC uses multiple techniques to ensure process isolation, such as for example namespaces, cgroups and capabilities. Use of the LXC containers 10 guarantees a level of isolation between various processes running inside the containers 10. Additionally, a connection inspector 20 provides transport level security by monitoring the packets dynamically.

One LXC container 10 is run for each untrusted solution. The container boundary serves as the trust boundary of the solution. The container 10 is started by a non-root user who has permission to mount a device storage partition on a pre-specified mount point created exclusively for the untrusted solutions. As a result, if and when a privilege escalation occurs, the container process cannot cause harm to the host device.

A set of subuids are used by each of the processes running inside the container. These allow the processes to be mapped to a parent process outside the container 10, in order to identify individual processes.

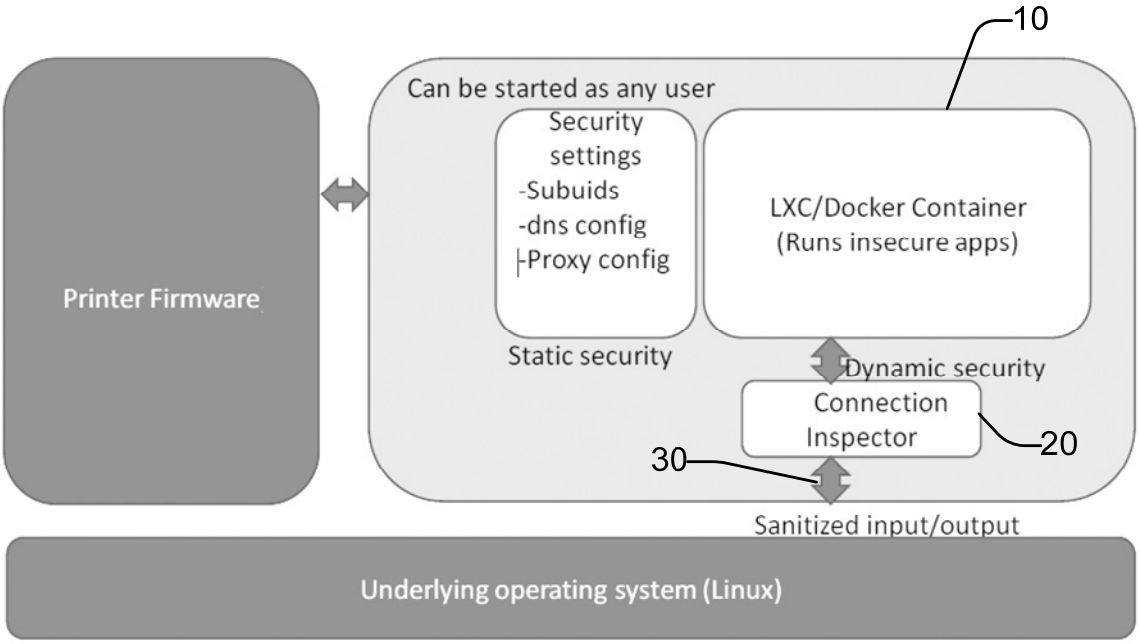
A network bridge 30 is created for the exclusive use of the container 10. The bridge 30 is configured as the only gate for all inbound and outbound communications. The bridge 30 is further configured to use a pre-defined DNS proxy, for the purpose of ensuring secure resolution of DNS queries, and to track the domains referred to by the containerized applications. Similarly, the HTTP proxy is also pre-configured to refer to a pre-defined process (the connection inspector 20), which logs all the inbound and outbound traffic. This enables online or offline analysis for validating the contents of the message.

The secure HTTP connection request is made from the HTTP proxy, not the containerized application, so that the HTTP packets can be tracked for security purposes.

The disclosed technique advantageously allows the manufacturer of an electronic device to ensure the security of partner applications and solutions embedded in the device. The time and effort for validating these partner solutions is reduced, and the partners obtain

increased flexibility to use their choice of libraries in developing their apps.

Disclosed by Raghu Anantharangachar, S. Chaitra, Upendra Bhangui Avadhut, Shankaranarayana Viswanatha, and Travis M. Cossel, Hewlett-Packard Inc.


10
Can be started as any user
Security settings
-Subuids
-dns config
-Proxy config
LXC/Docker Container
(Runs insecure apps)
Printer Firmware
Static security
Dynamic security
Connection Inspector
20
30
Sanitized input/output
Underlying operating system (Linux)