CLAIMS:

1. A system comprising: a graphics hardware component;
a host processor to execute an application lacking system level privileges;
a memory device to detect a triggering of a doorbell, to identify the application based on an address of the doorbell, and to issue a notification of a workload submission by the application;
a controller including, a notification module to receive the notification of the workload submission; and
a scheduler to schedule a transfer of the workload submission to the graphics hardware component for execution, wherein the transfer is to bypass an operating system associated with the host processor, and wherein the scheduler includes a parser to, parse the workload submission to obtain one or more commands, modify at least one of the one or more commands, add at least one command to the one or more commands, prioritize the one or more commands, and prevent one or more untrusted applications from submitting privileged commands to the graphics hardware component.

2. The system of claim 1, wherein the transfer is to bypass a kernel mode driver associated with the graphics hardware component.

3. The system of claim 1, wherein the notification is to be an interrupt.

4. The system of claim 1, wherein the scheduler further includes:
a retrieval component to retrieve the workload submission from a work queue associated with the application;
an enqueue component to add the one or more commands to a submission queue associated with the graphics hardware component; and
a dequeue component to transfer the one or more commands from the submission queue to the graphics hardware component in accordance with a graphics submission policy.

5. A method of operating a controller, comprising:
receiving, at the controller, a notification of a workload submission by an application lacking system level privileges, wherein the notification is issued by a memory device when detecting a triggering of a doorbell;
parsing the workload submission to obtain one or more commands, modifying at least one of the one or more commands, adding at least one command to the one or more commands, and prioritizing the one or more commands, wherein the parsing of the

workload submission includes preventing one or more untrusted applications from submitting privileged commands to a graphics hardware component; and

using the controller to schedule a transfer of the workload submission to the graphics hardware component for execution, wherein the controller has system level privileges, wherein the transfer bypasses an operating system.

6. The method of claim 5, wherein the transfer bypasses a kernel mode driver associated with the graphics hardware component.

7. The method of claim 5, wherein the notification is an interrupt.

8. The method of claim 5, wherein using the controller to schedule the transfer includes:

retrieving the workload submission from a work queue associated with the application;

adding the one or more commands to a submission queue associated with the graphics hardware component; and

transferring the one or more commands from the submission queue to the graphics hardware component in accordance with a graphics submission policy.

9. A controller comprising:

a notification module to receive a notification of a workload submission by an application lacking system level privileges, wherein the notification is issued by a memory device when detecting a triggering of a doorbell;

a parser to parse the workload submission to obtain one or more commands, modify at least one of the one or more commands, add at least one command to the one or more commands, and prioritize the one or more commands, wherein to parse the workload submission is to include a prevention of one or more untrusted applications from submitting privileged commands to a graphics hardware component; and

a scheduler to schedule a transfer of the workload submission to the graphics hardware component for execution, wherein the scheduler has system level privileges, wherein the transfer is to bypass an operating system.

10. The controller of claim 9, wherein the transfer is to bypass a kernel mode driver associated with the graphics hardware component.

11. The controller of claim 9, wherein the notification is to be an interrupt.

12. The controller of claim 9, wherein the scheduler includes:

a retrieval component to retrieve the workload submission from a work queue associated with the application;

an enqueue component to add the one or more commands to a submission queue associated with the graphics hardware component; and

a dequeue component to transfer the one or more commands from the submission queue to the graphics hardware component in accordance with a graphics submission policy.

13. At least one non-transitory computer readable storage medium comprising a set of instructions which, when executed by a controller, cause the controller to: receive a notification of a workload submission by an application lacking system level privileges, wherein the notification is issued by a memory device when detecting a triggering of a doorbell;

parse the workload submission to obtain one or more commands, modify at least one of the one or more commands, add at least one command to the one or more commands, and prioritize the one or more commands, wherein the parse of the workload submission includes a prevention of one or more untrusted applications from submitting privileged commands to a graphics hardware component; and

schedule a transfer of the workload submission to the graphics hardware component for execution, wherein the controller is to have system level privileges, wherein the transfer is to bypass an operating system.

14. The at least one non-transitory computer readable storage medium of claim 13, wherein the transfer is to bypass a kernel mode driver associated with the graphics hardware component.

15. The at least one non-transitory computer readable storage medium of claim 13, wherein the notification is to be a deferred procedure call.

16. The at least one non-transitory computer readable storage medium of claim 13, wherein the instructions, when executed, cause the controller to:

retrieve the workload submission from a work queue associated with the application;

add the one or more commands to a submission queue associated with the graphics hardware component; and

transfer the one or more commands from the submission queue to the graphics hardware component in accordance with a graphics submission policy.