kubernetes #72259 issue : #72124

commit : 704414592037dee57de1071cf8489373b6ddd5d0

问题（原因）：如果一个pod（podA）的nominateNodeName没有及时加上，那么scheduler就可能会将其他pod放置到该nominate node上，导致nominate node的资源被占用，从而导致podA无法被分配。因为podA无法被分配，进而导致调度队列中的其他pod无法被调度。

as the comments in #72124
Setting nominatedNodeName has an inherent race:

1. The scheduler tries a pod and finds it unschedulable.
2. The preemption logic tries and finds a node to schedule the pod (with or without preempting any other pod).
3. The scheduler sends a pod update to update nominatedNodeName of the pod. The scheduler has NOT received the pod update for setting "nominatedNodeName" yet.
4. The scheduler tries another pod and finds the same node schedulable. So, it binds the second pod.
5. The nominated Pod has the nominatedNodeName set, but the resources on the node are already taken by another pod, so the nominated pod is no longer schedulable on that node.

The race that I described above can cause an issue:

1. There is a pod which has nominatedNodeName, but is affected by the above race.
2. A smaller pod with the same priority is scheduled on the same node (due to the above race condition).
3. There is enough room on the node for other small pods, but since there exists a nominated pod for the node, other smaller pods won't be scheduled there. The nominated pod cannot be scheduled there either, because the available resources are partially taken by the other smaller pod and there are not enough resources for the nominated pod.

复现：无

修复：在向API server更新pod的nominateNodeName之前在scheduler中更新它对应的nominate node name

```
diff --git a/pkg/scheduler/scheduler.go b/pkg/scheduler/scheduler.go
index a9e2d12ec6..ce35a58cbc 100644
--- a/pkg/scheduler/scheduler.go
+++ b/pkg/scheduler/scheduler.go
@@ -320,11 +320,19 @@ func (sched *Scheduler) preempt(preemptor *v1.Pod,
scheduleErr error) (string, e
        var nodeName = ""
```

```
        if node != nil {
                nodeName = node.Name
+               // Update the scheduling queue with the nominated pod
information. Without
+               // this, there would be a race condition between the
next scheduling cycle
+               // and the time the scheduler receives a Pod Update for
the nominated pod.
+
sched.config.SchedulingQueue.UpdateNominatedPodForNode(preemptor,
nodeName)
+
+               // Make a call to update nominated node name of the pod
on the API server.
                err =
sched.config.PodPreemptor.SetNominatedNodeName(preemptor, nodeName)
                if err != nil {
                        klog.Errorf("Error in preemption process. Cannot
update pod %v/%v annotations: %v", preemptor.Namespace, preemptor.Name,
err)
+
sched.config.SchedulingQueue.DeleteNominatedPodIfExists(preemptor)
                        return "", err
                }
```