

```

1 public class SyncRequestProcessor {
2     public void run() {
3         while (running) {
4             if (logCount > (snapCount / 2))
5                 zks.takeSnapshot();
6             ...
7         }
8     }
9 }
10 public class DataTree {
11     public void serializeNode(OutputArchive oa, ...) {
12         ...
13         String children[] = null;
14         synchronized (node) {
15             scount++;
16             oa.writeRecord(node, "node");
17             children = node.getChildren();
18         }
19         ...
20     }
21 }

```

① identify long-running region

③ reduce

② locate vulnerable operations

④ insert context hooks

+ ContextManger.serializeNode\_reduced\_args\_setter(oa, node);

(a) A module in main program

```

1 public class SyncRequestProcessor$Checker {
2     public static void serializeNode_reduced(
3         OutputArchive arg0, DataNode arg1) {
4         arg0.writeRecord(arg1, "node");
5     }
6     public static void serializeNode_invoke() {
7         Context ctx = ContextManger.
8         serializeNode_reduced_context();
9         if (ctx.status == READY) {
10             OutputArchive arg0 = ctx.args_getter(0);
11             DataNode arg1 = ctx.args_getter(1);
12             serializeNode_reduced(arg0, arg1);
13         }
14     }
15     public static void takeSnapshot_reduced() {
16         serializeList_invoke();
17         serializeNode_invoke();
18     }
19     public static Status checkTargetFunction0() {
20         ...
21         takeSnapshot_reduced();
22     }
23 }

```

④ generate context factory

⑤ add fault signal checks

(b) Generated checker