TECHNIQUES FOR METADATA PROCESSING

BACKGROUND

This application relates generally to data processing and, more particularly, to programmable units for metadata processing.

Today's computer systems are notoriously hard to secure. Conventional processor architectures, for example, allow various behaviors, such as buffer overflows, pointer forging and the like, which violate higher level abstractions. Closing the gap between programming language and hardware may be left to software, where the cost of enforcing airtight abstractions is often deemed too high.

Some recent efforts have demonstrated the value of propagating metadata during execution to enforce policies that catch safety violations and malicious attacks as they occur. These policies can be enforced in software but typically incur high, undesirable overheads, such as in performance and/or costs, that discourage their deployment or other motivate coarse approximations providing less protection. Hardware support for fixed policies may reduce overhead to acceptable levels and prevent a large portion of undesired code violations, such as may be performed by malicious code or malware attacks. For example, Intel recently announced hardware for bounds checking and isolation. While these mitigate many of today's attacks, fully securing systems will require more than memory safety and isolation. Attacks rapidly evolve to exploit any remaining forms of vulnerability.

Claims

1. A method of storing tag values for a memory region comprising:

receiving one or more tag values for a plurality of memory locations in the memory region; and

creating a hierarchical representation of the one or more tag values for the plurality of memory locations of the memory region, the hierarchical representation including one or more nodes each associated with one or more memory locations, wherein each node of the hierarchical representation either denotes a tag value for a subrange of one or more memory locations associated with said each node, or otherwise denotes that said each node at a particular level of the hierarchical representation does not specify a tag value for the subrange of one or more memory locations associated with said each node whereby one or more other nodes at one or more levels lower than the particular level specify one or more tag values for the subrange of one or more memory locations associated with said each node.

2. The method of Claim 1, further comprising:

receiving a first tag value to be stored as a current tag value for a first memory location of the memory region; and

updating the hierarchical representation to denote that the first tag value is the current tag value for the first memory location.

3. The method of Claim 2, wherein, prior to performing said updating, an existing tag value is stored as the current tag value for the first memory location and the method includes performing processing to update the current tag value from the existing tag value to the first tag value, said processing comprising:

determining a first node of the hierarchical representation associated with a first subrange of memory locations including the first memory location, wherein the first node denotes the existing tag value as a homogenous tag value for the first subrange of

memory locations whereby each memory location in the first subrange has the existing tag value;

creating a first child node of the first node, the first child node denoting the first tag value as the current tag value for the first memory location of the first subrange;

creating one or more other child nodes of the first node, the one or more other child nodes denoting the existing tag value for memory locations of the first subrange other than the first memory location; and

updating the first node to indicate that the first node does not specify a tag value for the subrange of one or more memory locations associated with said first node whereby one or more other descendant nodes of the first node, at one or more levels of the hierarchical representation lower than a level of the first node, specify one or more tag values for the first subrange.

4. The method of Claim 1, further comprising performing first processing to determine a current tag value for a first memory location of the memory region, said first processing including:

traversing the hierarchical representation and locating a first node of the hierarchical representation associated with a first subrange of one or more memory locations of the memory region, the first subrange including the first memory location, wherein the first node denotes an existing tag value for the first subrange of one or more memory locations whereby each memory location in the first subrange has the existing tag value, wherein the existing tag value is the current tag value for the first memory location.

5. The method of Claim 4, wherein said traversing commences from a root node of the hierarchical representation and proceeds downward to lower levels of the hierarchical representation toward leaf nodes of the hierarchical representation.

6. The method of Claim 4, wherein the hierarchical representation includes a plurality

of levels having nodes of the plurality of levels stored in a plurality of caching levels, wherein nodes from each of the plurality of levels are stored in a different corresponding caching level of the plurality of caching levels.

7. The method of Claim 6, wherein said first processing includes looking up the first memory location in each of the plurality of caching levels to determine whether each of the caching levels includes a node defining the current tag value for the first memory location.

8. The method of Claim 7, wherein said looking up performs a parallel lookup of the plurality of caching levels to determine which of the plurality of caching levels defines the current tag value for the first memory location.

9. The method of Claim 8, wherein a first caching level of the plurality of caching levels specifies the current tag value for the first memory location, wherein the first node is included in the first caching level.

10. The method of Claim 9, wherein the first caching level is a highest of the plurality of levels of the hierarchy defining a tag value for the first memory location.

11. The method of Claim 1, wherein the hierarchical representation is a tree.

12. The method of Claim 11, wherein the tree includes one or more leaf nodes specifying current tag values for the plurality of memory locations of the memory region.

13. The method of Claim 11, wherein the tree is a binary tree.

14. The method of Claim 11, wherein each node of the tree has zero of more child

nodes up to a maximum allowable number of child nodes.

15. The method of Claim 1, wherein the hierarchical representation includes a plurality of levels having nodes of the plurality of levels stored in a plurality of caching levels, wherein nodes from each of the plurality of levels are stored in a different corresponding caching level of the plurality of caching levels, and the method further comprising:

determining whether there is a data cache hit or data cache miss for a first memory location of the memory region;

responsive to determining a data cache miss for the first memory location, performing first processing including looking up a cached tag value for the first memory location in the plurality of caching levels; and

responsive to determining a data cache hit for the first memory location, performing second processing including:

determining whether a first cached tag value as stored in the data cache for the first memory location matches a current tag value as stored in the hierarchical representation for the first memory location; and

if it is determined that the first cached tag value and the current tag value do not match, accordingly updating the first cached tag value as stored in the data cache.

16. The method of Claim 15, wherein updating the first cached tag value as stored in the data cache includes:

replacing the first cached tag value of the data cache with the current tag value for the first memory location as stored in the hierarchical representation; and

updating additional information maintained in the data cache denoting which of the plurality of caching levels includes the current tag value for the first memory location.

17. The method of Claim 15, wherein said first processing looks up the cached tag value for the first memory location in the plurality of caching levels by searching the plurality of caching levels in parallel.

18. The method of Claim 17, wherein, responsive to a tag cache miss when looking up the cached tag value for the first memory location, said tag cache miss being at a particular one of the plurality of caching levels where the particular one of the plurality of caching levels does not include tag value for the first memory location, processing performed includes:

traversing the hierarchical representation to locate a node specifying a tag value for the first memory location; and

inserting tag values for nodes of the hierarchical representation into appropriate ones of the plurality of caching levels in accordance with said traversing.

19. The method of Claim 15, wherein said first processing looks up the cached tag value for the first memory location in the plurality of caching levels by traversing the plurality of caching levels from a highest of the caching levels to a lowest of the caching levels

20. The method of Claim 19, wherein, responsive to a tag cache miss when looking up the cached tag value for the first memory location, said tag cache miss being at a particular one of the plurality of caching levels where the particular one of the plurality of caching levels does not include tag value for the first memory location, processing performed includes:

traversing the hierarchical representation to locate a node specifying a tag value for the first memory location; and

inserting one or more tag values for nodes of the hierarchical representation into appropriate ones of the plurality of caching levels in accordance with said traversing.

21. The method of Claim 1, wherein the tag values are metadata tags used in metadata processing.

22. A system comprising:

a processor; and

a memory comprising code stored thereon that, when executed, performs a method of storing tag values for a memory region comprising:

receiving one or more tag values for a plurality of memory locations in the memory region; and

creating a hierarchical representation of the one or more tag values for the plurality of memory locations of the memory region, the hierarchical representation including one or more nodes each associated with one or more memory locations, wherein each node of the hierarchical representation either denotes a tag value for a subrange of one or more memory locations associated with said each node, or otherwise denotes that said each node at a particular level of the hierarchical representation does not specify a tag value for the subrange of one or more memory locations associated with said each node whereby one or more other nodes at one or more levels lower than the particular level specify one or more tag values for the subrange of one or more memory locations associated with said each node.

23. A non-transitory computer readable medium comprising code stored thereon that, when executed, performs a method of storing tag values for a memory region comprising:

receiving one or more tag values for a plurality of memory locations in the memory region; and

creating a hierarchical representation of the one or more tag values for the plurality of memory locations of the memory region, the hierarchical representation including one or more nodes each associated with one or more memory locations,

wherein each node of the hierarchical representation either denotes a tag value for a subrange of one or more memory locations associated with said each node, or otherwise denotes that said each node at a particular level of the hierarchical representation does not specify a tag value for the subrange of one or more memory locations associated with said each node whereby one or more other nodes at one or more levels lower than the particular level specify one or more tag values for the subrange of one or more memory locations associated with said each node.