🖥 **FreeAndFair** / **ColoradoRLA**

<> Code        ⊙ Issues  81        ⅱ Pull requests  12        ▶ Actions        ▥ Projects  5        ⊘ Security

New issue                                                                                      **Jump to bottom**

# Consider Logging Integrity Issues #393

⊙ Open        **dmzimmerman** opened this issue on Aug 22, 2017 · 1 comment

| | |
|---|---|
| **Assignees** | 👤👤 |
| **Labels** | **feature**   security   server |
| **Projects** | ▥ system design and domain ...   ▥ server   ▥ quality assurance |
| **Milestone** | ⬗ someday |

---

🔴 **dmzimmerman** commented on Aug 22, 2017                                     Contributor

Right now, the logs in the database are hashchained - you can pick any log entry, follow the chain of hashes back to the root, and know that no entry in between has been tampered with. However, it is possible, through the magic of optimistic locking and concurrent transactions, for the log to become a tree rather than a single linear sequence. In that scenario, one could potentially remove an entire branch of the tree without it being detectable in an investigation of the logs.

Possible solutions for this include:

- implementing an actual Merkle tree, as compared to the upside down one we have now - computationally very costly with each transaction
- checking at every log commit to see if there are in fact multiple branches, and if so, adding a "merge" log entry to tie the branches together before committing new logs (not clear how well this would actually work, though, and it still gives a window, albeit very small, for log tampering)
- ???

👤 🔴 **dmzimmerman** assigned **dmzimmerman** and **kiniry** on Aug 22, 2017

🏷 🔴 **dmzimmerman** added   security   server   labels on Aug 22, 2017

🏳  👤 **kiniry** added this to the **someday** milestone on Aug 25, 2017

---

👤 **kiniry** commented on Aug 25, 2017                                      [ Contributor ]

I do not think that this issue is significant enough to raise with CDOS for this first release of the tool. We do need to think about and solve it, and I'm hoping we'll get that opportunity in a future contract.

---

🏷  👤 **kiniry** added the   `feature`   label on Aug 25, 2017

▥  👤 **kiniry** added this to **Todo** in **quality assurance** on Aug 26, 2017

▥  👤 **kiniry** added this to **Todo** in **server** on Aug 26, 2017

▥  👤 **kiniry** added this to **Todo** in **system design and domain engineering** on Aug 26, 2017

## Assignees

👤 **dmzimmerman**

👤 **kiniry**

---

## Labels

**feature**   security   server

---

## Projects

▥  **system design and domain engineering**
    Todo

▥  **server**
    Todo

▥  **quality assurance**
    Todo

---

## Milestone

someday

---

## Linked pull requests

Successfully merging a pull request may close this issue.

None yet

---

## 2 participants