

## FreeNAS - Feature #37363

### Add API call for returning configuration database

07/10/2018 08:59 AM - Ken Moore

<b>Status:</b> Done	<b>Estimated time:</b> 0.00 hour
<b>Priority:</b> No priority	
<b>Assignee:</b> Vladimir Vinogradenko	
<b>Category:</b> Middleware	
<b>Target version:</b> 11.3-ALPHA1	
<b>Severity:</b> Medium	<b>Needs Merging:</b> No
<b>Reason for Closing:</b>	<b>Needs Automation:</b> No
<b>Reason for Blocked:</b>	<b>Support Suite Ticket:</b> n/a
<b>Needs QA:</b> No	<b>Hardware Configuration:</b>
<b>Needs Doc:</b> No	

**Description**

Needed for TrueView:  
Add an API call which will return the entire system configuration settings/database.

This will be used to generate an audit trail by comparing the settings before/after a user has directly accessed a FreeNAS system.

It would be nice if the settings were reported as a JSON data object for easier parsing/comparison later - but that is not essential.

--- William's edit -----  
We need a new API call called `datastore.dump_json` where it will go through all models/tables as an array.  
Each item of the array will be comprised of:

- table\_name
- verbose\_name
- fields
- name
- verbose\_name
- database\_type
- entries

key: val

**Risk**  
This offer low risk since this is just an API addition

**Acceptance Criteria**  
Calling `midclt call datastore.dump_json`` should provide a full dump of the database in JSON format.

#### Associated revisions

##### Revision 0d67e8dc - 07/12/2018 08:16 AM - Vladimir Vinogradenko

`datastore.dump_json`

Ticket: #37363

##### Revision c71c7828 - 07/12/2018 08:20 AM - Vladimir Vinogradenko

`datastore.dump_json`

Ticket: #37363

## History

---

### #1 - 07/10/2018 09:23 AM - Dru Lavigne

- Assignee changed from Release Council to William Grzybowski

### #2 - 07/10/2018 10:13 AM - William Grzybowski

- Status changed from Unscreened to Blocked

- Reason for Blocked set to Waiting for feedback

Is this really the job of TrueView to do audit trail?

There is no guarantee of database schema changes between versions, neither guarantees changes didnt happen between begin and end of session (e.g. admin logged in, made a change that reflected database, performed some action based on that changed, deleted the change from database, admin logged out).

### #3 - 07/11/2018 05:25 AM - Ken Moore

Yes - TrueView is supposed to be providing an audit trail of changes by individual administrators.

Since FreeNAS only has the "root" user and does not provide time-based audit logs, the idea to work around that limitation was to have TrueView fetch the configuration database immediately before/after an admin requests direct access to the box (TrueView already limits direct access requests to specific subsystems of the angular UI), and then look at the changes during that session (with a limit of 1 admin session at-a-time - perhaps per-subsystem).

Perhaps a better way to do this (but will require much more work in the FreeNAS Middleware) is to actually save a time-based audit trail which can then be retrieved by TrueView based on the time period of the access or fetch the audit of all changes associated with a particular auth session token.

### #4 - 07/11/2018 06:20 AM - William Grzybowski

Ken Moore wrote:

Yes - TrueView is supposed to be providing an audit trail of changes by individual administrators.

Since FreeNAS only has the "root" user and does not provide time-based audit logs, the idea to work around that limitation was to have TrueView fetch the configuration database immediately before/after an admin requests direct access to the box (TrueView already limits direct access requests to specific subsystems of the angular UI), and then look at the changes during that session (with a limit of 1 admin session at-a-time - perhaps per-subsystem).

Perhaps a better way to do this (but will require much more work in the FreeNAS Middleware) is to actually save a time-based audit trail which can then be retrieved by TrueView based on the time period of the access or fetch the audit of all changes associated with a particular auth session token.

I agree that would be a much better and bullet proof way. Database snapshot is not a guarantee of anything, operations can be made that do not affect database and as explained it leaves holes.

However that is a big project for middleware on our side. What is the urgency of this feature? Can you live without it for some time?

**#5 - 07/11/2018 06:25 AM - Ken Moore**

This was one of the primary features of TrueView that was requested by Sales/Management (RBAC - Role based access control, must include audit trail/logs). That is why I created the ticket for the database dump rather than a full audit trail within FreeNAS itself - that is a fairly easy (but incomplete) solution for this issue that can be replaced by a more expansive solution later on.

**#6 - 07/11/2018 06:39 AM - William Grzybowski**

- Description updated
- Status changed from Blocked to Unscreened
- Assignee changed from William Grzybowski to Vladimir Vinogradenko
- Target version changed from Backlog to 11.3
- Severity changed from New to Medium
- Reason for Blocked deleted (Waiting for feedback)
- Needs Doc changed from Yes to No
- Needs Merging changed from Yes to No

**#7 - 07/11/2018 07:04 AM - William Grzybowski**

- Description updated

**#8 - 07/12/2018 04:20 AM - Vladimir Vinogradenko**

- Status changed from Unscreened to In Progress

**#9 - 07/12/2018 09:37 AM - Vladimir Vinogradenko**

- Status changed from In Progress to Ready for Testing

<https://github.com/freenas/freenas/pull/1528>

**#10 - 01/21/2019 11:00 AM - Dru Lavigne**

- Target version changed from 11.3 to 11.3-BETA1

**#12 - 02/12/2019 07:38 AM - Ken Moore**

Verified that the datastore.dump\_json API call works via the console on an 11-nightlies VM.

**#13 - 02/12/2019 07:56 AM - Dru Lavigne**

- Status changed from Ready for Testing to Done
- Needs QA changed from Yes to No

**#14 - 04/02/2019 08:21 AM - Dru Lavigne**

- Target version changed from 11.3-BETA1 to 11.3-ALPHA1