Foreman - Bug #34093

Authorizer cache loads all permitted resources to memory

12/06/2021 12:28 PM - Tomer Brisker

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Tomer Brisker</td>
</tr>
<tr>
<td>Category:</td>
<td>Users, Roles and Permissions</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>Difficulty:</td>
<td></td>
</tr>
<tr>
<td>Triaged:</td>
<td>No</td>
</tr>
<tr>
<td>Bugzilla link:</td>
<td></td>
</tr>
<tr>
<td>Pull request:</td>
<td><a href="https://github.com/theforeman/foreman/pull/8972">https://github.com/theforeman/foreman/pull/8972</a></td>
</tr>
<tr>
<td>Fixed in Releases:</td>
<td>3.1.0</td>
</tr>
<tr>
<td>Found in Releases:</td>
<td></td>
</tr>
</tbody>
</table>

Description
The cache saves calculating the scope for the collection, but loads all records when calling `.include?` instead of checking for the existence of the specific record within the scope

```ruby
def collection_cache_lookup(subject, permission)
  collection = @cache[subject.class.to_s][permission] ||=
    find_collection(subject.class, :permission => permission)

  collection.include?(subject)
end
```

Related issues:
Related to Foreman - Bug #33585: Puma memory usage after update to 2.5 - Closed

Associated revisions

Revision 700f3a69 - 12/08/2021 03:30 PM - Tomer Brisker
Fixes #34093 - Don't eager load all authorized resources

The authorizer cache currently loads all permitted resources to memory when `.includes?` is called just to check if one resource is in the list. This can get very heavy when authorizing resources that have many records, for example ConfigReports.
Instead of caching the scope and loading all AR objects to memory, we can cache just the ids to reduce the memory usage of the cache. This is still sub-optimal for resources that have millions of records, but is faster for pages such as index pages which perform multiple permissions checks for every item and don't need to perform an additional sql query for every check.

Revision 92d40b23 - 12/08/2021 03:30 PM - Tomer Brisker
Refs #34093 - Minor authorizer clean ups

  `.any?` avoids triggering another sql query for loaded relations, but in
In this specific case the relation won't be loaded. `.present?` loads the relation to an array and checks if it is empty, which is obviously worse than checking in SQL.
- Use regular `Hash` instead of `HashWithIndefferentAccess`.
  HWIA comes with a performance overhead since it needs to handle both string and symbol keys. Since the `@cache` is only used in one place, we can control the keys and use a regular Hash instead.

Revision 5e3686a5 - 12/08/2021 03:30 PM - Tomer Brisker
Refs #34093 - Allow skipping cache from `User.can?`.

Revision cbe90240 - 12/08/2021 03:30 PM - Tomer Brisker
Refs #34093 - Improve host config status performance.

- Skip the authorizer cache. There can be millions of ConfigReports in the db. there is no need to load all of them just to check if the user can see the last one.
- Cache the status_link method result. The status_link method checks for the user's permissions and is called 3 times - twice in `host_detailed_status_list` directly and once inside `host_global_status_icon_class`. We can avoid the extra permissions check by caching the method's results.

Revision 806d9324 - 12/08/2021 03:30 PM - Tomer Brisker
Refs #34093 - Document Authorizer#can? and log huge cache loads.

Also flatten the authorizer cache instead of a nested hash.

History

#1 - 12/06/2021 12:30 PM - Tomer Brisker
- Description updated

#2 - 12/06/2021 12:33 PM - The Foreman Bot
- Status changed from New to Ready For Testing
- Pull request https://github.com/theforeman/foreman/pull/8972 added

#3 - 12/08/2021 03:30 PM - The Foreman Bot
- Fixed in Releases 3.2.0 added

#4 - 12/08/2021 04:01 PM - Tomer Brisker
- Status changed from Ready For Testing to Closed

Applied in changeset foreman|700f3a69bebe0753092a9db1f6be6a03ebed7df0f.
#5 - 12/08/2021 04:19 PM - Tomer Brisker
- Fixed in Releases 3.1.0 added
- Fixed in Releases deleted (3.2.0)

#6 - 12/20/2021 08:19 AM - Lukas Zapletal
- Related to Bug #33585: Puma memory usage after update to 2.5 added