# Pulp - Issue #6025

## Pulp 3.0 remote cert sha256 value doesn't match the input or the cert

01/19/2020 05:31 PM - binlinf0

| | | | |
|---|---|---|---|
| **Status:** | CLOSED - CURRENTRELEASE | **Start date:** | 01/19/2020 |
| **Priority:** | Normal | **Due date:** | |
| **Assignee:** | lmjachky | **Estimated time:** | 0:00 hour |
| **Category:** | | | |
| **Sprint/Milestone:** | 3.2.0 | | |
| **Severity:** | 2. Medium | **Groomed:** | No |
| **Version:** | | **Sprint Candidate:** | No |
| **Platform Release:** | | **Tags:** | |
| **OS:** | | **Sprint:** | Sprint 66 |
| **Triaged:** | Yes | **Quarter:** | |

**Description**

When I list a remote, the certs are showed like below
{
"ca_cert": "b8bd944ff40f1756c08743800453b724f133029725dc762ed9ce6504a828a5ec",
"client_cert": "d24baa45b4b554e782dd134e6c5f1eb1f88e1d88122d3d417b05f7f8153954a1",
"client_key": "55ece66e807b5c979f425e55f90958b8fbc769320a0bdb5f4c3563464e8c9530",
...
}

Tied to compare the hexdigest from the ca_cert input string and using openssl command give the same sha256 sum but not match when query pulp.
hashlib.sha256(bytes(ca_cert_string, "utf8")).hexdigest()

1. openssl dgst -sha256 redhat-uep.pem
   SHA256(redhat-uep.pem)= 39e65fabe7560d366be3bc4d133bcdef13e30d41ac552a05d182e2f66395422d

This prevent us from comparing the current cert on the system with the cert configured in the pulp. The sha256 sum don't match even the cert is working and configured correctly.

### Associated revisions

**Revision 34dcd641 - 02/19/2020 08:54 PM - Lubos Mjachky**

Do not trim leading and trailing whitespace characters

In this commit, there was also fixed the documentation. The field itself returns a hash of a file content, not a hash of a certificate stored within a file.

closes #6025 https://pulp.plan.io/issues/6025

### History

**#1 - 01/21/2020 04:32 PM - fao89**

*- Triaged changed from No to Yes*

*- Sprint set to Sprint 64*

**#2 - 01/23/2020 09:22 PM - jsherril@redhat.com**

*- File test2.crt added*

This is working fine for me, but the key thing is to remove all trailing whitespace before trying to compute it. Here's the ruby code we're using to do the comparison:

```
Digest::SHA256.hexdigest(computed_options[key].chomp)
```

I've uploaded a file that when sent to pulp3 as a remote client cert, comes back with the checksum of 6066b56511a9fabaea2afe86af8a91c75ad2d37bf31891bd493eb8c54f45583d

This file is pre-chomp'd

#### #3 - 01/24/2020 02:58 PM - rchan

*- Sprint changed from Sprint 64 to Sprint 65*

#### #4 - 02/01/2020 05:55 PM - binlinf0

Your test crt works fine. I just cannot figure out why the redhat-uep.pem on the host doesn't match the sha256sum. I don't see any trailing white space in the pem. Here is my test program and the pem file

https://github.com/bli111/pulp_test/blob/master/remote.py

$ ./remote.py INPUT 39e65fabe7560d366be3bc4d133bcdef13e30d41ac552a05d182e2f66395422d PULP client_cert
b8bd944ff40f1756c08743800453b724f133029725dc762ed9ce6504a828a5ec

#### #5 - 02/07/2020 03:20 PM - rchan

*- Sprint changed from Sprint 65 to Sprint 66*

#### #6 - 02/12/2020 10:00 PM - lmjachky

*- Description updated*

*- Status changed from NEW to ASSIGNED*

*- Assignee set to lmjachky*

#### #7 - 02/14/2020 07:37 PM - lmjachky

I confirm that the issue you observed is caused by trailing whitespace characters. In the file redhat-uep.pem, there is present the newline character at the end of the file (even if you do not see it). On the other hand, the file test2.crt does not contain any trailing whitespace characters at all.

This behavior is caused by the implementation of CharField which is inherited by SecretCharField, and the field client_cert is an instance of SecretCharField. CharField removes trailing whitespace characters by default (https://www.django-rest-framework.org/api-guide/fields/#charfield).

```
    def __init__(self, **kwargs):
        self.allow_blank = kwargs.pop('allow_blank', False)
        self.trim_whitespace = kwargs.pop('trim_whitespace', True)
```

```
    def to_internal_value(self, data):
        # We're lenient with allowing basic numerics to be coerced into strings,
        # but other types should fail. Eg. unclear if booleans should represent as `true` or `True`,
        # and composites such as lists are likely user error.
        if isinstance(data, bool) or not isinstance(data, (str, int, float,)):
            self.fail('invalid')
        value = str(data)
        return value.strip() if self.trim_whitespace else value
```

We can disable the trimming by declaring trim_whitespace=False in the corresponding serializer. For now, I can only recommend you to remove all trailing whitespace characters until we reach a consensus:

```
cert = f.read().rstrip()
```

#### #8 - 02/19/2020 01:09 PM - lmjachky

*- Status changed from ASSIGNED to POST*

https://github.com/pulp/pulpcore/pull/550

#### #9 - 02/20/2020 10:05 PM - Anonymous

*- Status changed from POST to MODIFIED*

Applied in changeset pulpcore|34dcd64175a7f19b236b6a555985eb613aefb736.

#### #10 - 02/27/2020 06:46 PM - daviddavis

*- Status changed from MODIFIED to CLOSED - CURRENTRELEASE*

#### #11 - 02/27/2020 07:15 PM - daviddavis

*- Sprint/Milestone set to 3.2.0*

#### #12 - 06/16/2020 10:54 PM - bmbouter

*- Category deleted (14)*

We are removing the 'API' category per open floor discussion June 16, 2020.

**Files**

| | | | |
|---|---|---|---|
| test2.crt | 30.4 KB | 01/23/2020 | jsherril@redhat.com |