


Unwrapped orphan renewable token, never has its ttl/lease duration incremented #2134

New issue

 Closed

 bitsofinfo opened this issue on Nov 28, 2016 · 11 comments



bitsofinfo commented on Nov 28, 2016 • edited ▼

I read the docs and just as an average user attempting to create a wrapped token that will be renewed, and this doesn't seem to work work.

Doesn't make much sense to me, every iteration of renewing the token results in the `lease_duration` never incrementing (always decreasing) and the token eventually expires... resulting in a 403 permission denied...

The `{VAULT_CLIENT_TOKEN}` used below to create the orphan wrapped token has a 1 hour long TTL... but since the token it creates is an orphan I don't see how that should matter

```
#!/bin/bash -x

read -p "Enter Username: " VAULT_USER
read -s -p "Enter Password: " VAULT_PW

VAULT_ADDR=$1
POLICY_NAME=$2

# Auth the invoking user
authResult=$(curl -s ${1}/v1/auth/ldap/login/${VAULT_USER} -d '{"password": "${VAULT_PW}" }')
VAULT_CLIENT_TOKEN=$(echo $authResult | sed -e 's/^.*"client_token": "[^"]*"$/\1/'`

# Generate wrapped token for the app
tokenCreateResult=$(curl -s -X POST -H "X-Vault-Token: ${VAULT_CLIENT_TOKEN}" -H "X-Vault-Wrap-TTL: 5m" -d '{"renewable":true,"ttl":"1m","no_parent":true,"policies":["default","${POLICY_NAME}"]}' ${VAULT_ADDR}/v1/auth/token/create-orphan)
VAULT_WRAPPED_TOKEN=$(echo $tokenCreateResult | sed -e 's/^.*"token": "[^"]*"$/\1/'`

# Unwrap token
unwrapTokenResult=$(curl -s -X POST -H "X-Vault-Token: ${VAULT_WRAPPED_TOKEN}" ${VAULT_ADDR}/v1/sys/wrapping/unwrap)
VAULT_UNWRAPPED_TOKEN=$(echo $unwrapTokenResult | sed -e 's/^.*"client_token": "[^"]*"$/\1/'`

echo
echo $VAULT_UNWRAPPED_TOKEN

while true; do

    lookupResult=$(curl -s -X GET -H "X-Vault-Token: ${VAULT_UNWRAPPED_TOKEN}" ${VAULT_ADDR}/v1/auth/token/lookup-self)
    LOOKUP_RESULT=$(echo $lookupResult`

    renewResult=$(curl -s -X POST -H "X-Vault-Token: ${VAULT_UNWRAPPED_TOKEN}" ${VAULT_ADDR}/v1/auth/token/renew-self)
    RENEW_RESULT=$(echo $renewResult`
    sleep 30s

done
```

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

0.6.3


Notifications

2 participants




 This was referenced on Nov 28, 2016

vault token-renew <token> <increment> doesn't increase the time remaining on a token TTL. #1079

 Closed

0.18 renewals of wrapped token error with 403 after 1 hour despite renewals and orphan hashicorp/consul-template#806

 Closed



bitsofinfo commented on Nov 28, 2016 • edited ▼

Note if I pass an `increment` of "1m", then it seems to properly renew it (the `ttl/lease_duration` bounces from 29 to 60 as expected)

Without the `increment` parameter in the `renew-self` it fails eventually w/ the 403/expiry as the `lease_duration` never increases. The documentation states `increment` is optional



jefferai commented on Nov 28, 2016

Owner

I'm a bit confused. Wrapped tokens can't be renewed, and in your example above you're not attempting to renew a wrapped token, you're attempting to renew a normal token that you create using the first token that you got from LDAP.

In addition you say that you are creating an hour long token, but your request actually creates a 1 minute long TTL, not an hour (`ttl=1m`).

Are you saying that if you don't wrap the token initially, and simply call the `userpass` authentication endpoint without requesting wrapping, the token that comes back behaves differently?

There isn't much else I can tell you without more information -- what version of Vault, what is your LDAP configuration, what is the output of `vault auth -methods` ?

🔖 jefferai added this to the **0.6.3** milestone on Nov 28, 2016



bitsofinfo commented on Nov 28, 2016

I'm a bit confused. Wrapped tokens can't be renewed, and in your example above you're not attempting to renew a wrapped token, you're attempting to renew a normal token that you create using the first token that you got from LDAP.

Correct. I am trying to renew the **unwrapped** token, not the wrapped token as that is not possible

In addition you say that you are creating an hour long token, but your request actually creates a 1 minute long TTL, not an hour (`ttl=1m`).

No, I'm just noting that the token being used to invoke `auth/token/create-orphan` has an 1 hour long TTL itself. (i.e. if inheritance is in question, which it shouldn't be because the created token is an orphan)

We are running vault 0.6.2

You can see all the output by just running that sample bash script above.

```
$ vault auth -methods
Path    Type  Default TTL  Max TTL  Description
ldap/    ldap  system      system
token/  token system      system  token based credentials

$ vault read auth/ldap/config
Key      Value
---      -
binddn    uid=whatever,ou=dept,dc=mycompany,dc=org
bindpass  xxxx
certificate
discoverdn false
groupattr hasAccessRole
groupdn   ou=people,dc=mycompany,dc=org
groupfilter (&(objectClass=myPersonClass)(uid={{.Username}}))
insecure_tls true
starttls  true
tls_min_version tls12
upndomain
url       ldap://openldap.mycompany.org
userattr  email
userdn    ou=people,dc=mycompany,dc=org
```



jefferai commented on Nov 28, 2016

Owner

I can't really replicate this:

```
$ vault token-create -orphan -ttl=1h
Key      Value
---      -
token    fcadebd4-8b73-1274-b6df-a432db67f261
```

```
token_accessor bd0c397a-468c-0816-82b2-cfadd6e29edb
token_duration 1h0m0s
token_renewable true
token_policies [root]

$ VAULT_TOKEN=fcadebd4-8b73-1274-b6df-a432db67f261 vault token-create -orphan -ttl=1m -
policy=default
Key            Value
---            -
token          fbca8db2-660c-e59b-9ffe-e233de4201e0
token_accessor 876a22d8-a35e-daa4-8109-eabaddc64c63
token_duration 1m0s
token_renewable true
token_policies [default]

$ VAULT_TOKEN=fbca8db2-660c-e59b-9ffe-e233de4201e0 vault token-renew
Key            Value
---            -
token          fbca8db2-660c-e59b-9ffe-e233de4201e0
token_accessor 876a22d8-a35e-daa4-8109-eabaddc64c63
token_duration 767h59m46s
token_renewable true
token_policies [default]
```

Wrapping wouldn't have anything to do with this, and this is all normal token handling stuff. If you don't give an increment when renewing self, it should fall back to your system defaults since you don't have any tuning on the auth mount.



bitsofinfo commented on Nov 28, 2016

Please don't use the CLI, but the script I pasted above that does all calls via REST



jefferai commented on Nov 28, 2016

Owner

@bitsofinfo Conversely, please use the CLI because it a) makes it much easier to create reproducible and understandable test cases and b) is itself nothing more than an API client for the REST interface that is maintained by us. If the CLI works and your script doesn't, it makes it much clearer on which call the issue might lie.



bitsofinfo commented on Nov 28, 2016

I'm not installing the vault client in all our docker containers.



jefferai commented on Nov 28, 2016

Owner

I wouldn't think you'd need to. Just run the CLI during debugging, and if erroneous REST calls are discovered, they can be fixed.

Alternately, change your script to something that can display that behavior against a fresh Vault config and without requiring your LDAP sever, e.g. `vault server -dev` using just the token backend.

As I cannot replicate this behavior otherwise, it's up to you.



bitsofinfo commented on Nov 28, 2016

Yes I get the exact same result as you do with the basic example you pasted above (i.e. it works), but unfortunately that is not what we are doing.

We are using the LDAP auth backend and the REST api to create the tokens and renew them.

Please see below.

Basically what happens is that we create a token, then continually look it up, renew it for "1m", sleep 15s. This works fine and the `lookup-self` calls reflect this state properly over and over.... however no-matter what, after about 1 hour of iterations the renewals no longer have any effect and the lease_duration goes to zero and we get a 403

Auth request, wrapped token creation and unwrapping

AUTH REQUEST

```
curl -s https://myvault:8200/v1/auth/ldap/login/my.user@mycompany.com -d '{ "password": "123" }'
```

AUTH RESPONSE

```
{
  "request_id": "1019e1d8-37f9-76b2-581d-5b43184528c9",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": {

  },
  "wrap_info": null,
  "warnings": null,
  "auth": {
    "client_token": "XXXXXX-5559-b317-70d9-b227fa9a775f",
    "accessor": "XXXXXX-abd8-16ca-2b1a-XXXXXXX",
    "policies": [
      "default",
      "vaultadminuser"
    ],
    "metadata": {
      "policies": "default,vaultadminuser",
      "username": "my.user@mycompany.com"
    },
    "lease_duration": 3600,
    "renewable": true
  }
}
```

WRAPPED TOKEN CREATE

```
curl -s -X POST -H 'X-Vault-Token: XXXXXX-5559-b317-70d9-b227fa9a775f' -H 'X-Vault-Wrap-TTL: 5m'
-d '{ "renewable": true, "ttl": "1m", "no_parent": true, "policies": ["default", "my-policy"] }'
https://myvault:8200/v1/auth/token/create-orphan
```

WRAPPED TOKEN CREATE RESPONSE

```
{
  "request_id": "",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": null,
  "wrap_info": {
    "token": "bc9e35ab-b666-7c7f-7e8c-33eabe16103f",
    "ttl": 300,
    "creation_time": "2016-11-28T21:59:35.841210127Z",
    "wrapped_accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1"
  },
  "warnings": null,
  "auth": null
}
```

UNWRAP TOKEN REQUEST

```
curl -s -X POST -H 'X-Vault-Token: bc9e35ab-b666-7c7f-7e8c-33eabe16103f'
https://myvault:8200/v1/sys/wrapping/unwrap
```

UNWRAP TOKEN RESPONSE

```
{
  "request_id": "40dc54f6-dec2-60a2-11dc-e3b34f7d4110",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": null,
  "wrap_info": null,
}
```

```
"warnings": null,
"auth": {
  "client_token": "2f1adeb0-2cb3-6a0d-534b-cb9bbea4dc78",
  "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
  "policies": [
    "default",
    "my-policy"
  ],
  "metadata": null,
  "lease_duration": 60,
  "renewable": true
}
}
```

Everything below repeats every 15 seconds (lookup, renew, lookup, sleep...)

First iteration

LOOKUP TOKEN

```
curl -s -X GET -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbea4dc78'
https://myvault:8200/v1/auth/token/lookup-self
```

LOOKUP TOKEN RESPONSE

```
{
  "request_id": "e5132a6b-b9eb-6650-5492-dd36aff56ee7",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": {
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
    "creation_time": 1480370375,
    "creation_ttl": 60,
    "display_name": "token",
    "explicit_max_ttl": 0,
    "id": "2f1adeb0-2cb3-6a0d-534b-cb9bbea4dc78",
    "meta": null,
    "num_uses": 0,
    "orphan": true,
    "path": "auth/token/create-orphan",
    "policies": [
      "default",
      "my-policy"
    ],
    "renewable": true,
    "ttl": 58
  },
  "wrap_info": null,
  "warnings": null,
  "auth": null
}
```

RENEW TOKEN

```
curl -s -X POST -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbea4dc78' -d '{"increment":"1m"}'
https://myvault:8200/v1/auth/token/renew-self
```

RENEW TOKEN RESPONSE

```
{
  "request_id": "196e8456-83a8-999a-119a-e4a225c08fe4",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": null,
  "wrap_info": null,
  "warnings": null,
  "auth": {
    "client_token": "2f1adeb0-2cb3-6a0d-534b-cb9bbea4dc78",
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
    "policies": [
      "default",
      "my-policy"
    ]
  }
}
```

```
    ],
    "metadata": null,
    "lease_duration": 60,
    "renewable": true
  }
}
```

LOOKUP TOKEN

```
curl -s -X GET -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78'
https://myvault:8200/v1/auth/token/lookup-self
```

LOOKUP TOKEN RESPONSE

```
{
  "request_id": "1e5062c6-4bc1-ca93-d552-85257de70396",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": {
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
    "creation_time": 1480370375,
    "creation_ttl": 60,
    "display_name": "token",
    "explicit_max_ttl": 0,
    "id": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",
    "last_renewal_time": 1480370377,
    "meta": null,
    "num_uses": 0,
    "orphan": true,
    "path": "auth/token/create-orphan",
    "policies": [
      "default",
      "my-policy"
    ],
    "renewable": true,
    "ttl": 59
  },
  "wrap_info": null,
  "warnings": null,
  "auth": null
}
```

Sample random iteration about 30 minutes into it...

LOOKUP TOKEN

```
curl -s -X GET -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78'
https://myvault:8200/v1/auth/token/lookup-self
```

LOOKUP TOKEN RESPONSE

```
{
  "request_id": "5a8ec5fe-ed1e-400c-673a-a2f49ff22bdc",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": {
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
    "creation_time": 1480370375,
    "creation_ttl": 60,
    "display_name": "token",
    "explicit_max_ttl": 0,
    "id": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",
    "last_renewal_time": 1480371457,
    "meta": null,
    "num_uses": 0,
    "orphan": true,
    "path": "auth/token/create-orphan",
    "policies": [
      "default",
      "my-policy"
    ],
    "renewable": true,
    "ttl": 43
  },
  "wrap_info": null,
  "warnings": null,
}
```

```
"auth": null
}
```

RENEW TOKEN

```
curl -s -X POST -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78' -d '{"increment":"1m"}'
https://myvault:8200/v1/auth/token/renew-self
```

RENEW TOKEN RESPONSE

```
{
  "request_id": "6c9fcf24-6014-b4e5-ba28-1aa032612263",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": null,
  "wrap_info": null,
  "warnings": null,
  "auth": {
    "client_token": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
    "policies": [
      "default",
      "my-policy"
    ],
    "metadata": null,
    "lease_duration": 60,
    "renewable": true
  }
}
```

LOOKUP TOKEN

```
curl -s -X GET -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78'
https://myvault:8200/v1/auth/token/lookup-self
```

LOOKUP TOKEN RESPONSE

```
{
  "request_id": "1c202cc5-b1e3-a52d-4237-371611e89bd3",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": {
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
    "creation_time": 1480370375,
    "creation_ttl": 60,
    "display_name": "token",
    "explicit_max_ttl": 0,
    "id": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",
    "last_renewal_time": 1480371474,
    "meta": null,
    "num_uses": 0,
    "orphan": true,
    "path": "auth/token/create-orphan",
    "policies": [
      "default",
      "my-policy"
    ],
    "renewable": true,
    "ttl": 60
  },
  "wrap_info": null,
  "warnings": null,
  "auth": null
}
```

As the script approaches one hour of runtime within that loop of lookup->renew->lookup. Note the `renew-self` call is NOT incrementing the time anymore.

LOOKUP TOKEN

```
curl -s -X GET -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78'  
https://myvault:8200/v1/auth/token/lookup-self
```

LOOKUP TOKEN RESPONSE

```
{  
  "request_id": "c94c34d7-c57f-4277-d578-a91299bd06c2",  
  "lease_id": "",  
  "renewable": false,  
  "lease_duration": 0,  
  "data": {  
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",  
    "creation_time": 1480370375,  
    "creation_ttl": 60,  
    "display_name": "token",  
    "explicit_max_ttl": 0,  
    "id": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",  
    "last_renewal_time": 1480373916,  
    "meta": null,  
    "num_uses": 0,  
    "orphan": true,  
    "path": "auth/token/create-orphan",  
    "policies": [  
      "default",  
      "my-policy"  
    ],  
    "renewable": true,  
    "ttl": 43  
  },  
  "wrap_info": null,  
  "warnings": null,  
  "auth": null  
}
```

RENEW TOKEN

```
curl -s -X POST -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78' -d '{"increment":"1m"}'  
https://myvault:8200/v1/auth/token/renew-self
```

RENEW TOKEN RESPONSE

```
{  
  "request_id": "0a97a63c-654f-0277-5b24-40489e12509b",  
  "lease_id": "",  
  "renewable": false,  
  "lease_duration": 0,  
  "data": null,  
  "wrap_info": null,  
  "warnings": null,  
  "auth": {  
    "client_token": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",  
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",  
    "policies": [  
      "default",  
      "my-policy"  
    ],  
    "metadata": null,  
    "lease_duration": 43,  
    "renewable": true  
  }  
}
```

LOOKUP TOKEN

```
curl -s -X GET -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78'  
https://myvault:8200/v1/auth/token/lookup-self
```

LOOKUP TOKEN RESPONSE

```
{  
  "request_id": "03506d1c-cb4d-319e-3374-68821b47866d",  
  "lease_id": "",  
  "renewable": false,  
  "lease_duration": 0,  
  "data": {  
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",  
    "creation_time": 1480370375,  
    "creation_ttl": 60,  
    "display_name": "token",  
    "explicit_max_ttl": 0,  
    "id": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",  
    "last_renewal_time": 1480373916,  
    "meta": null,  
    "num_uses": 0,  
    "orphan": true,  
    "path": "auth/token/create-orphan",  
    "policies": [  
      "default",  
      "my-policy"  
    ],  
    "renewable": true,  
    "ttl": 43  
  },  
  "wrap_info": null,  
  "warnings": null,  
  "auth": null  
}
```



```

"creation_time": 1480370375,
"creation_ttl": 60,
"display_name": "token",
"explicit_max_ttl": 0,
"id": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",
"last_renewal_time": 1480373932,
"meta": null,
"num_uses": 0,
"orphan": true,
"path": "auth/token/create-orphan",
"policies": [
  "default",
  "my-policy"
],
"renewable": true,
"ttl": 42
},
"wrap_info": null,
"warnings": null,
"auth": null
}

```

The last few iterations, the TTL continues to decrease, and NOT be incremented and we finally get a 403

```

curl -s -X POST -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78' -d '{"increment":"1m"}'
https://myvault:8200/v1/auth/token/renew-self

```

```

{
  "request_id": "9ac1b3df-be3b-469e-2a5a-7538a6773309",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": null,
  "wrap_info": null,
  "warnings": null,
  "auth": {
    "client_token": "2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78",
    "accessor": "6296e963-dc3c-cdde-4cf3-564b95d26dd1",
    "policies": [
      "default",
      "my-policy"
    ],
    "metadata": null,
    "lease_duration": 10,
    "renewable": true
  }
}

```

```

curl -s -X GET -H 'X-Vault-Token: 2f1adeb0-2cb3-6a0d-534b-cb9bbee4dc78'
https://myvault:8200/v1/auth/token/lookup-self

```

```

{"errors":["permission' 'denied']}

```



jefferai commented on Nov 28, 2016

Owner

I'm not really able to reconcile what you posted above with your earlier statement:

Note if I pass an increment of "1m", then it seems to properly renew it (the ttl/lease_duration bounces from 29 to 60 as expected)

Without the increment parameter in the renew-self it fails eventually w/ the 403/expiry as the lease_duration never increases. The documentation states increment is optional

In your pasted code above this is not the behavior. You are passing an increment and it is not having an effect. This is what I would expect.

You're hitting a max TTL somewhere. My guess, since the output of `vault auth -methods` showed the LDAP mount using system defaults, is that you've set the default max TTL in your config file to be one hour. In the absence of any override on the mount this will then cap your token lifetime to one hour. You may want to read <https://www.vaultproject.io/docs/concepts/tokens.html> which talks about TTLs and max TTLs extensively.



bitsofinfo commented on Nov 29, 2016

So you are correct, turns out this is related to two things

- `max_lease_ttl` of 1 hour in the vault server config the container I am hitting was setup with
- The token created by the script was not `periodic` causing it to be bound by that `max_lease_ttl` no matter what I did w/ renewals.

Note that the `period` option is described in the CLI documentation but not on the token create REST endpoint documentation, please add this, or would you like a separate ticket for that. Once i started passing the `period` option on the token create REST calls, then this all started working as expected with token's living beyond one hour.

Docs here, need the *parameters* section updated for info about `period`

<https://www.vaultproject.io/docs/auth/token.html>

`/auth/token/create`

`/auth/token/create-orphan`

Note if I pass an increment of "1m", then it seems to properly renew it (the `ttl/lease_duration` bounces from 29 to 60 as expected)

Correct, within the 1 hour max lease ttl period, it was seemingly working, but only as it reached the `max_lease_ttl` set on the server level in the configs (1hour) it stopped working.

 jefferai closed this on Dec 1, 2016