Arvados - Feature #15881

[controller] LDAP login support

11/27/2019 03:28 PM - Peter Amstutz

Status:	Resolved	**Start date:**	05/06/2020
Priority:	Normal	**Due date:**	
Assigned To:	Tom Clegg	**% Done:**	100%
Category:	Login	**Estimated time:**	0.00 hour
Target version:	2020-06-03 Sprint		

Description

Discussed at engineering meeting 2020 April 15, using PAM to access LDAP is unlikely to be satisfactory since it doesn't provide any user profile information. Add first-class LDAP support to the new username/password authenticate endpoint.

Subtasks:

Task # 16350: Determine set of features required to support existing LDAP users	**Closed**
Task # 16351: Review 15881-ldap	**Resolved**
Task # 16467: Review 15881-ldap	**Resolved**

Related issues:

Related to Arvados Epics - Story #15322: Replace and delete sso-provider	**Resolved**	**03/11/2020**	**08/26/2020**
Related to Arvados - Story #16453: [controller] Expand config comment about L...	**New**		
Related to Arvados - Bug #16475: Javascript mystery investigation on WB2 work...	**Resolved**		
Has duplicate Arvados - Story #15883: Support LDAP logins	**Duplicate**		

Associated revisions

Revision 374cc9ff - 05/14/2020 05:38 PM - Tom Clegg

Merge branch '15881-ldap'

refs #15881

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <tom@tomclegg.ca>

Revision 8e504f3c - 05/18/2020 09:03 PM - Lucas Di Pentima

Merge branch '15881-ldap'
Refs #15881

Arvados-DCO-1.1-Signed-off-by: Lucas Di Pentima <lucas@di-pentima.com.ar>

Revision 46db403c - 05/27/2020 08:10 PM - Tom Clegg

Merge branch '15881-ldap'

refs #15881

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <tom@tomclegg.ca>

History

#1 - 12/18/2019 03:46 PM - Peter Amstutz

- Has duplicate Story #15883: Support LDAP logins added

#2 - 12/18/2019 04:16 PM - Peter Amstutz

- Related to Story #15322: Replace and delete sso-provider added

#3 - 12/31/2019 11:13 PM - Peter Amstutz

- Category set to Login

#4 - 03/04/2020 05:02 PM - Peter Amstutz

- Target version changed from To Be Groomed to 2020-03-25 Sprint

#5 - 03/04/2020 05:26 PM - Peter Amstutz

- Blocked by Feature #16212: Can choose PAM as an authentication backend added

#6 - 03/11/2020 03:31 PM - Tom Clegg

- Assigned To set to Tom Clegg

#7 - 03/25/2020 01:43 PM - Peter Amstutz

- Target version changed from 2020-03-25 Sprint to 2020-04-08 Sprint

#8 - 03/25/2020 01:48 PM - Peter Amstutz

- Status changed from New to In Progress

#9 - 04/08/2020 03:35 PM - Tom Clegg

- Target version changed from 2020-04-08 Sprint to 2020-04-22

#10 - 04/15/2020 09:21 PM - Peter Amstutz

- Description updated

#11 - 04/15/2020 09:21 PM - Peter Amstutz

- Blocked by deleted (Feature #16212: Can choose PAM as an authentication backend)

#12 - 04/16/2020 07:05 PM - Peter Amstutz

- Target version changed from 2020-04-22 to 2020-05-06 Sprint

#13 - 05/06/2020 03:30 PM - Tom Clegg

- Target version changed from 2020-05-06 Sprint to 2020-05-20 Sprint

#14 - 05/07/2020 01:57 PM - Tom Clegg

15881-ldap @ bbb132e983f9ec5c7d50cf0ab709ec041af1f844 -- https://ci.arvados.org/view/Developer/job/developer-run-tests/1836/

#15 - 05/07/2020 03:13 PM - Tom Clegg

- File 15881-doc.png added

#16 - 05/07/2020 05:06 PM - Tom Clegg

LDAP has several configs and it seemed sensible to put them in an LDAP section. But this means we have

```
Login:
  # ...
  GoogleClientID: ""
  GoogleClientSecret: ""
  GoogleAlternateEmailAddresses: true
  PAM: true
  PAMService: arvados
  LDAP:
    Enable: true
    URL: ldap://ldap:389
    # ...
```

Seems like we should pick one approach here: all flat, or all hierarchical like this:

```
Login:
  # ...
  Google:
    Enable: true
    ClientID: ""
    ClientSecret: ""
    AlternateEmailAddresses: true
  PAM:
    Enable: true
    Service: arvados
  LDAP:
    Enable: true
    URL: ldap://ldap:389
    # ...
```

#17 - 05/07/2020 05:15 PM - Peter Amstutz

Tom Clegg wrote:

> LDAP has several configs and it seemed sensible to put them in an LDAP section. But this means we have
>
> [...]
>
> Seems like we should pick one approach here: all flat, or all hierarchical like this:
>
> [...]

I am fine with migrating to a more hierarchical layout, and we already have a concept of deprecated configs that are automatically migrated.

However, this also affects the exported config, so anything that's expecting the old organization breaks. We don't yet have a general solution for that.

Now on its own, probably the only thing that is affected is Workbench 2 reading the PAM config. However this is the sort of thing I was trying to avoid by proposing a "LoginEndpoint" field or some other strategy for that Workbench 2 can use to determine how to do login without having to know about every login method support by Arvados.

Also, Workbench 2 needs to support LDAP, so I think we need to address this anyway.

#18 - 05/07/2020 06:26 PM - Tom Clegg

> However, this also affects the exported config, so anything that's expecting the old organization breaks.

Fortunately Login.PAM is the only config here that's ever been exported, and it has only existed as an experimental feature in dev/prerelease versions, so I think we're safe.

#19 - 05/08/2020 05:54 PM - Tom Clegg

- File 15881-docs.png added

Set up web based login

Select one of the following login mechanisms for your cluster.

1. If all users will authenticate with Google, configure Google login.
2. If all users will authenticate with an existing LDAP service, configure LDAP.
3. If all users will authenticate using PAM as configured on your controller node, configure PAM.
4. If you need to enable multiple authentication methods, configure a separate single sign-on (SSO) server.

Google login

With this configuration, users will sign in with their Google accounts.

First, visit Setting up Google auth.

Next, enable Google login and copy the values of **Client ID** and **Client secret** from the Google Developers Console into the `Login.Google` section of `config.yml`:

```
    Login:
      Google:
        Enable: true
        ClientID: "0000000000000-zzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzz.apps.googleusercontent.com"
        ClientSecret: "zzzzzzzzzzzzzzzzzzzzzz"
```

LDAP

With this configuration, authentication uses an external LDAP service like OpenLDAP or Active Directory.

Enable LDAP authentication and provide your LDAP server's host, port, and credentials (if needed to search the directory) in `config.yml`:

```
    Login:
      LDAP:
        Enable: true
        URL: ldap://ldap.example.com:389
        SearchBindUser: cn=lookupuser,dc=example,dc=com
        SearchBindPassword: xxxxxxxx
        SearchBase: ou=Users,dc=example,dc=com
```

The email address reported by LDAP will be used as primary key for Arvados accounts. This means **users must not be able to edit their own email addresses** in the directory.

Additional configuration settings are available:

- `StartTLS` is enabled by default.
- `StripDomain` and `AppendDomain` modify the username entered by the user before searching for it in the directory.
- `SearchAttribute` (default `uid`) is the LDAP attribute used when searching for usernames.
- `SearchFilters` accepts LDAP filter expressions to control which users can log in.

Check the LDAP section in the default config file for more details and configuration options.

PAM (experimental)

With this configuration, authentication is done according to the Linux PAM ("Pluggable Authentication Modules") configuration on your controller host.

Enable PAM authentication in `config.yml`:

```
    Login:
      PAM:
        Enable: true
```

Check the default config file for more PAM configuration options.

The default PAM configuration on most Linux systems uses the local password database in `/etc/shadow` for all logins. In this case, in order to log in to Arvados, users must have a shell account and password on the controller host itself. This can be convenient for a single-user or test cluster.

PAM can also be configured to use different backends like LDAP. In a production environment, PAM configuration should use the service name ("arvados" by default) to set a separate policy for Arvados logins: generally, Arvados users should not have shell accounts on the controller node.

For information about configuring PAM, refer to the PAM System Administrator's Guide.

Separate single-sign-on (SSO) server

With this configuration, Arvados passes off authentication to a separate SSO server that supports Google, LDAP, and a local password database.

See Install the Single Sign On (SSO) server

#20 - 05/08/2020 05:55 PM - Tom Clegg

15881-ldap @ 0634b763dd27914cff5ca49c6cfe11233746ee31 -- https://ci.arvados.org/view/Developer/job/developer-run-tests/1842/

#21 - 05/12/2020 08:49 PM - Lucas Di Pentima

Some comments & questions:

- Documentation
  - Would it be convenient to warn the reader that SSO's going to be decommissioned in the near future? I guess new installations shouldn't be encouraged to think that multiple auth methods will be something supported through SSO.
- Configuration
  - Do you think we could maintain config naming consistency by renaming InsecureTLS to Insecure, or just use TLS.Insecure? Maybe the latter isn't convenient because TLS.Insecure is about Arvados' services.
  - Arvbox config needs an update on the Login section.
- File lib/controller/localdb/login_ldap.go - Line 92: Would it be better to do this check before any LDAP server interaction?
- File services/api/config/initializers/omniauth_init.rb - Lines 12 & 18 refers to deprecated Rails.configuration.Login["ProviderAppID"] config
- On 8f435f4bac86e7ba7dbd9770d2db9bb4db6cf569, the comment is: test LDAP login using a fake LDAP server, but I'm not able to understand how this is done, as the code updates don't seem to be about that. Is the godap dependency providing the LDAPSuite? A comment explaining how it works would be useful.
- Is the docker test being run by Jenkins? I'm not seeing any logging about it on "developer-run-tests-remainder" so I suppose it isn't, do you think it should be run every time? A least on my side, it isn't too slow, took like 20 seconds.

Other than that, LGTM.

#22 - 05/13/2020 07:43 PM - Tom Clegg

> - Would it be convenient to warn the reader that SSO's going to be decommissioned in the near future? I guess new installations shouldn't be encouraged to think that multiple auth methods will be something supported through SSO.

Yes, I've removed SSO entirely from the install guide, and added a "remove sso-provider" item to the future 2.1 upgrade notes.

> - Do you think we could maintain config naming consistency by renaming InsecureTLS to Insecure, or just use TLS.Insecure? Maybe the latter isn't convenient because TLS.Insecure is about Arvados' services.

Not sure about this. Some thoughts

- Even where TLS.Insecure == true (internal communication among Arvados components is unsecured) is acceptable, revealing users' LDAP credentials to a MitM could be a real problem. So I think it should be a separate knob.
- I agree symmetry with the other Insecure flag would be nice - but I also like that InsecureTLS makes it clear that we're talking about TLS, not other aspects of authentication that could be considered insecure, like empty/easy passwords.

> - Arvbox config needs an update on the Login section.

Updated.

> - File lib/controller/localdb/login_ldap.go - Line 92: Would it be better to do this check before any LDAP server interaction?

Yes, moved up.

> - File services/api/config/initializers/omniauth_init.rb - Lines 12 & 18 refers to deprecated Rails.configuration.Login["ProviderAppID"] config

Updated.

> - On 8f435f4bac86e7ba7dbd9770d2db9bb4db6cf569, the comment is: test LDAP login using a fake LDAP server, but I'm not able to understand how this is done, as the code updates don't seem to be about that. Is the godap dependency providing the LDAPSuite? A comment explaining how it works would be useful.

Oops, I missed adding the actual _test.go file in that commit. Fixed.

> - Is the docker test being run by Jenkins? I'm not seeing any logging about it on "developer-run-tests-remainder" so I suppose it isn't, do you think it should be run every time? A least on my side, it isn't too slow, took like 20 seconds.

It was disabled by default (needed go test -tags docker) but yes, it seems quick enough. I've enabled it.

15881-ldap @ 28e68f813bd7c48847c39a9e07d66ff5cf61662d -- https://ci.arvados.org/view/Developer/job/developer-run-tests/1850/

#23 - 05/13/2020 08:13 PM - Tom Clegg

> It was disabled by default (needed go test -tags docker) but yes, it seems quick enough. I've enabled it.

Oops, jenkins workers don't have docker. Now skipping the docker tests if docker info fails.

15881-ldap @ ca1eb648712232558014d648939868b2a902558a -- https://ci.arvados.org/view/Developer/job/developer-run-tests/1852/

#24 - 05/13/2020 09:48 PM - Lucas Di Pentima

Some minor documentation comments:

- File doc/admin/migrating-providers.html.textile.liquid - L16: typo “intead"
- The websocket installation doc page has a duplicated "Restart the API server and controller” section (one of which mentions the SSO)
- Doc file install/install-components.html.textile.liquid can be removed as it seems to have been replaced with install/install-manual-prerequisites.html.textile.liquid (also found because of SSO being mentioned)

And with that, it LGTM. Thanks!

#25 - 05/14/2020 05:35 PM - Tom Clegg

15881-ldap @ f26e4dec04ed9d68a9f826f72b4a9e627ddc4a5c -- https://ci.arvados.org/view/Developer/job/developer-run-tests/1856/

#26 - 05/14/2020 06:42 PM - Tom Clegg

I tried making the obvious changes to workbench2:

15881-ldap @ arvados-workbench2|6d2e6d292161d566f54e94f048805569ede8e3d5

I've managed to run unit tests, but not integration tests or actual browser tests.

#27 - 05/14/2020 10:12 PM - Lucas Di Pentima

Updates at arvados-workbench2|366e40d9
Test run: https://ci.arvados.org/view/Developer/job/developer-tests-workbench2/27/

Simplified and somehow fixed the code deciding if it should show the login form (integration tests were failing):

```
diff --git a/src/views/login-panel/login-panel.tsx b/src/views/login-panel/login-panel.tsx
index ba0f584f..f60f032a 100644
--- a/src/views/login-panel/login-panel.tsx
+++ b/src/views/login-panel/login-panel.tsx
@@ -11,6 +11,7 @@ import { ArvadosTheme } from '~/common/custom-theme';
 import { RootState } from '~/store/store';
 import { LoginForm } from '~/views-components/login-form/login-form';
 import Axios from 'axios';
+import { Config } from '~/common/config';

 type CssRules = 'root' | 'container' | 'title' | 'content' | 'content__bolder' | 'button';

@@ -69,6 +70,13 @@ type LoginPanelProps = DispatchProp<any> & WithStyles<CssRules> & {
     passwordLogin: boolean,
 };

+const requirePasswordLogin = (config: Config): boolean => {
+    if (config && config.clusterConfig) {
+        return config.clusterConfig.Login.LDAP.Enable || config.clusterConfig.Login.PAM.Enable || false;
+    }
+    return false;
+};
+
 export const LoginPanel = withStyles(styles)(
     connect((state: RootState) => ({
         remoteHosts: state.auth.remoteHosts,
@@ -76,10 +84,8 @@ export const LoginPanel = withStyles(styles)(
         localCluster: state.auth.localCluster,
         loginCluster: state.auth.loginCluster,
         welcomePage: state.auth.config.clusterConfig.Workbench.WelcomePageHTML,
-        passwordLogin: state.auth.remoteHostsConfig[state.auth.loginCluster || state.auth.homeCluster] &&
-
```

```
            state.auth.remoteHostsConfig[state.auth.loginCluster || state.auth.homeCluster].clusterConfig.Login.LDAP.Enable ||
-
            state.auth.remoteHostsConfig[state.auth.loginCluster || state.auth.homeCluster].clusterConfig.Login.PAM.Enable || false,
-
    }))(({ classes, dispatch, remoteHosts, homeCluster, localCluster, loginCluster, welcomePage, passwordLogin }: LoginPanelProps) => {
+
        passwordLogin: requirePasswordLogin(state.auth.remoteHostsConfig[state.auth.loginCluster || state.auth.homeCluster]),
+
        }))(({ classes, dispatch, remoteHosts, homeCluster, localCluster, loginCluster, welcomePage, passwordLogin }: LoginPanelProps) => {
         const loginBtnLabel = `Log in${(localCluster !== homeCluster && loginCluster !== homeCluster) ? " to "+localCluster+" with user from "+homeCluster : ''}`;

         return (<Grid container justify="center" alignItems="center"
```

It seems that reevaluating state.auth.remoteHostsConfig[state.auth.loginCluster || state.auth.homeCluster] sometimes returned undefined and made the code break.

Also, fixed the trailing slashes issue on URLs from #16392, so this makes wb2 work with arvbox again.

#28 - 05/15/2020 05:15 PM - Tom Clegg

> Simplified and somehow fixed the code deciding if it should show the login form (integration tests were failing):

Much better, thanks.

> It seems that reevaluating state.auth.remoteHostsConfig[state.auth.loginCluster || state.auth.homeCluster] sometimes returned undefined and made the code break.

That is a bit mysterious, isn't it?

> Also, fixed the trailing slashes issue on URLs from #16392, so this makes wb2 work with arvbox again.

Excellent, thanks. LGTM.

#29 - 05/20/2020 03:33 PM - Tom Clegg

- Related to Story #16453: [controller] Expand config comment about LDAP search filters added

#30 - 05/20/2020 03:44 PM - Tom Clegg

- Target version changed from 2020-05-20 Sprint to 2020-06-03 Sprint

#31 - 05/21/2020 08:26 PM - Tom Clegg

15881-ldap @ 95d08c91f6d902054eb9ed4f79cb6bda2c3e8342

Expands comment on SearchFilters config.

#32 - 05/27/2020 08:09 PM - Peter Amstutz

Tom Clegg wrote:

> 15881-ldap @ 95d08c91f6d902054eb9ed4f79cb6bda2c3e8342
>
> Expands comment on SearchFilters config.

LGTM.

#33 - 05/29/2020 08:28 PM - Tom Clegg

- Status changed from In Progress to Resolved

#34 - 05/29/2020 08:33 PM - Lucas Di Pentima

- Related to Bug #16475: Javascript mystery investigation on WB2 workaround added

#35 - 10/07/2020 02:11 AM - Peter Amstutz

- Release set to 25

Files

15881-doc.png	76.5 KB	05/07/2020	Tom Clegg
15881-docs.png	252 KB	05/08/2020	Tom Clegg