Contents

Extension/LDAPAuthentication

LDAP means Lightweight Directory Access Protocol. This can be used for centralized authentication. This extension allows you to connect BlueSpice (and MediaWiki) to an LDAP server for central authentication.

Important! Due to a security vulnerability in Active Directory, the use of a secure connection (with SSL / TLS encryption) for user authentication is strongly recommended. A Microsoft patch for March 2020 no longer allows unsecured connections:

https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/ADV190023

https://support.microsoft.com/en-us/help/4520412/2020-ldap-channel-binding-and-ldap-signing-requirement-for-windows

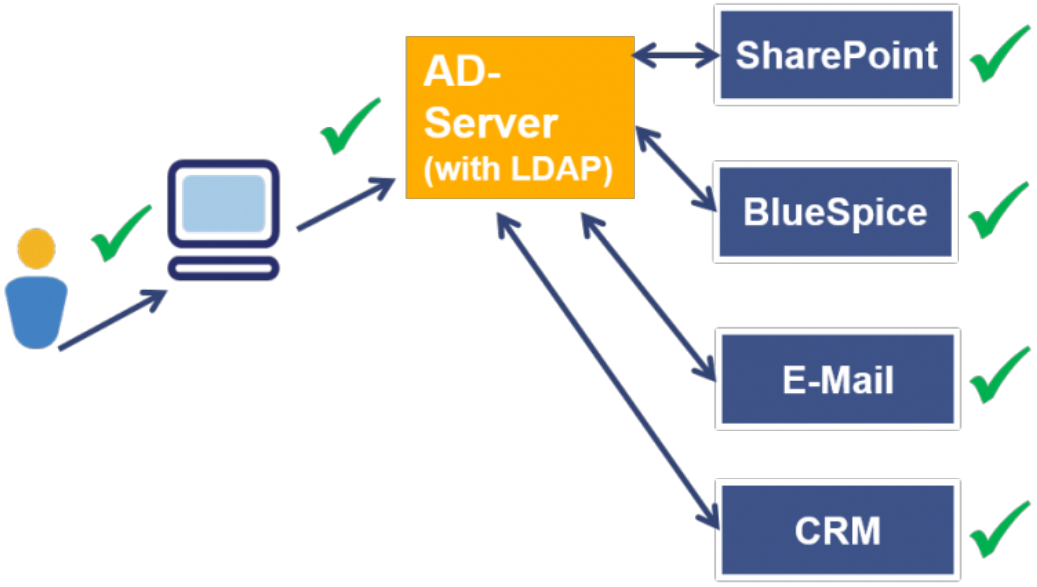
Any software that makes LDAP queries over unsecured connections is affected. Depending on the settings, this may also affect BlueSpice MediaWiki. If your system is configured to already use a secure connection, there is nothing you need to do.

If BlueSpice and Active Directory communicate via an unsecured connection, the system must be reconfigured to remain accessible.
Please check your configuration and contact us if you have any questions.

Contents

BlueSpice with LDAP

The MediaWiki extension "LDAP Authentication" is available in BlueSpice free, but is not activated by default. To link the Wiki to LDAP you need to activate the extension and configure it.



Configuration options

LDAP	Connection to AD without group synchronization
LDAP with group synchronization	If you have already defined user groups for your company, they can be adopted for the wiki. The corresponding groups are automatically present in the wiki with the group name and you can assign the corresponding rights to the groups there.
Comfort Sign-on	such as LDAP with group synchronization as above To further extend the connection to a central authentication, there is the possibility to set up a single sign-on. This means that the user is also logged on to the wiki at the same time when logging on to the PC.

Please note after configuring LDAP

- The wiki does not write back to the LDAP directory.
  That means, e.g. password changes in the wiki can lead to a conflict with the AD. In the best case, the changes will be overwritten by LDAP during the next login.
- It is not allowed to create users manually in the wiki. This leads to the conflict, even if the convention of the LdapAuthentication is case-sensitive.
- No users can be created in the LDAP directory via UserManager in the wiki.
- By default, there is no initial and / or active synchronization with the LDAP directory. Users do not appear until after the first login in the wiki.
- A group assignment is possible via the wiki.

LDAP configuration with group synchronization also applies:

- Again, the AD is the leading authority, Groups are taken over from the AD. Caution: The corresponding group must be created with the identical name, as it is called in the AD directory, in the group manager of the wiki in order to guarantee the group assignment.
- Groups can not be assigned to the user manually in the wiki. Please make the assignment via the LDAP directory.
- Groups that a user does not belong to in the LDAP directory are not displayed in the wiki.
- The assignment of the user to groups takes place during the login routine with the respective user, also here: no automatic comparison with the directory in the background.
- A group assignment via the wiki is not possible. Exceptions are the groups sysop, bot and bureaucrat. These can be assigned via the wiki and are not withdrawn from the user.

LDAP configuration with Comfort Sign-on also applies:

- Browser requirements: compatability with Internet Explorer, Edge und Google Chrome
- Security requirement: https, not http
- The web page must be assigned according to the local intranet (group guideline)

Related info

- Reference:LDAP_Authentication

Manual:Extension/LDAPAuthentication/LDAP

/etc/ldapprovider.json

ldapprovider.json

```
{
        "DOMAIN OF CUSTOMER": {
                "connection": {
                        "server": "",
                        "user": "",
                        "pass": "",
                        "basedn": "",
                        "userbasedn": "",
                        "groupbasedn": "",
                        "searchattribute": "samaccountname",
                        "usernameattribute": "samaccountname",
                        "realnameattribute": "displayname",
                        "emailattribute": "mail",
                        "grouprequest": "MediaWiki\\Extension\\LDAP
                        "nestedgroups": true
                },
                "authorization": {
                        "rules": {
                                "groups": {
                                        "required": [ "" ]
                                }
                        }
                },
                "userinfo": {
                        "attributes-map": {
                                "email": "mail",
                                "realname": "displayname",
                        }
                },
                "groupsync": {
                        "mechanism": "allgroups"
                }
        }
}
```

090-LDAP.php

```
wfLoadExtensions( [
        'Auth_remoteuser',// only needed if SingleSignOn is used
        'LDAPProvider',
        'Manual:Extension/LDAPAuthentication2',
        'LDAPAuthorization',
```

```
        'LDAPGroups',
        'LDAPUserInfo',
        'PluggableAuth'
] );

$LDAPProviderDomainConfigs = "/etc/ldapprovider.json";
$Manual:Extension/LDAPAuthentication2AllowLocalLogin = false;
$Manual:Extension/LDAPAuthentication2UsernameNormalizer = 'strtolow
$LDAPProviderCacheTime = 300;
$LDAPProviderCacheType = CACHE_MEMCACHED; // or CACHE_NONE if no me
$LDAPAuthorizationAutoAuthRemoteUserStringParser = 'username-at-dom

$bsgPermissionConfig['autocreateaccount'] = [ 'type' => 'global', "

$wgAuthRemoteuserUserName = function() {
        global $wgDBname;
        $user = '';
        if( isset( $_SERVER[ 'REMOTE_USER' ] ) ) {
                $user = $_SERVER[ 'REMOTE_USER' ];
        }

        //Bypass fot Parsoid / PhantomJS calls
        if( isset( $_SERVER[ 'REMOTE_ADDR' ] ) && substr( $_SERVER[
                if( empty( $user ) ) {
                        // check the 304f3058RemoteToken name of yo
                        $user = $_COOKIE[$wgDBname.'304f3058RemoteT
                }
        }

        return $user;
};
```

Note: This part *304f3058* of the cookie will change on some reasons. It should not, but we've seen it. So check with the cookies in your browser. $wgDBname is set in your LocalSettings.php

Install Single Sign on with Kerberos at CentOS 7.5.1408

Contents

About this manual

In this manual we will use certain placeholders. Replace them due to the following steps analogical to your system environment.

- `example.local` = your domain name
- `webserver.example.local` = FQDN of your bluespice webserver
- `dc.example.local` = FQDN of your domain controller

Preparation

Please make sure that you configured a working "A" record at your DNS server for `webserver.example.local` **and** (really necessary!) an reverse DNS record (PTR). Please make also sure that the system clocks do not differ more than 5 minutes.

Needed users in the Active Directory

You need to create the following users in your Active Directory

- One user for your Kerberos authentication (We will call it "KerberosProxy" at this manual)
- One user for your BlueSpice AD proxy user (We will call it "LdapProxy" at this manual)

Please create these users and configure the passwords to "never expire".

Create a keytab file

Create a keytab file at your domain controller using this command (works on Windows >= 2018 R2):

```
$ ktpass -princ HTTP/webserver.example.local@EXAMPLE.LOCAL
-mapuser KerberosProxy@EXAMPLE.LOCAL
-crypto RC4-HMAC-NT
-ptype KRB5_NT_PRINCIPAL
-pass <password-of-KerberosProxy>
-out bluespice.keytab
```

Move this file to your BlueSpice server (folder `/etc`).

Install required packages on CentOS

Install all packages you need for Kerberos:

```
$ yum install krb5-workstation mod_auth_kerb
```

Create Kerberos configuration

Create a backup of `/etc/krb5.conf` and clear the file content. Insert this new content:

```
[libdefaults]
         default_realm = EXAMPLE.LOCAL

[realms]
         EXAMPLE.LOCAL = {
                 kdc = dc.example.local
                 admin_server = dc.example.local
         }

[domain_realm]
         example.local = EXAMPLE.LOCAL
         .example.local = EXAMPLE.LOCAL
```

Test authentication with your Keytab-file

Now you can test your authentication with the keytab file which was created before:

```
$ kinit -VV -k -t /etc/bluespice.keytab HTTP/webserver.example.loca
```

If everything is configured correctly you should get a success message:

```
Authenticated to Kerberos v5
```

Secure your BlueSpice webroot with Kerberos

Now you have to secure the BlueSpice DocumentRoot with. Open your VirtualHost config and insert the following:

```
<VirtualHost *:443>
        ...
        <Directory /path/to/DocumentRoot>
                AuthType Kerberos
                KrbAuthRealms EXAMPLE.LOCAL
                KrbServiceName HTTP/webserver.example.local@EXAMPLE
                Krb5Keytab "/etc/bluespice.keytab"
                KrbMethodNegotiate on
                KrbMethodK5Passwd on
                Require valid-user
        </Directory>
        ...
</VirtualHost>
```

Restart your apache2 webserver.

Testing

Create a file `test.php` at the DocumentRoot of BlueSpice and insert this code:

```
<?php
echo $_SERVER['REMOTE_USER'];
```

If everyting works fine you should be able to open `test.php` with a webbrowser (not Firefox!) without getting an authentication window and you can see your windows user name at the `test.php`. Now delete `test.php`.

User permission to autocreate account

Make sure that the wiki user group `*` has the `autocreateaccount` permission in the PermissionManager of BlueSpice.

PHP-extension for ldap

Make sure that `php-ldap` is installed and loaded at your apache2 server.