Server preparation - Windows

installation, java, tomcat, quickstart, apache httpd

This tutorial shows you how to prepare the server for test or production use of CzechIdM. If you are looking for a much quicker way of installing CzechIdM, use the demo setup described here Getting Started

Basic system setup

- 1 server (can be virtualized) for everything: backend, frontend and database.
- OS Windows, ideally W2012 and newer
- PostgreSQL - installed from OpenSCG
- Java - installed from Oracle JDK
- Apache Tomcat - installed by Tomcat .exe installer
- Services start via system services (services.msc)

Instalation and software configuration

Prerequisities - Basic installation of Windows Server 2012.

- Install the **Telnet Client** system feature through **Programs and Features**. This is optional but greatly helps with debugging network problems.
- Install Firefox. Also optional, but greatly helps with debugging IdM webapp problems.
- Install Git Bash. This will be essential when configuring IdM and checking its logs.
  - Use **checkout-windows, commit unix style endings**. It does not really matter, we will not use the git command for anything.:)
  - **Do not** enable integration with windows cmd.
- Disable unnecessary windows services.
- Disable Microsoft IIS if installed.

PostgreSQL

On Windows, we use OpenSCG PostgreSQL distribution, version at least 9.6.

- Leave locations at default.
- Make sure you check the option to install the windows service.

Open the elevated shell (right-click on cmd and select "run as admin") and install the pgAdmin:

```
cd c:/postgresql
pgc install pgadmin3
```

If your server does not have Internet access, you can download and install pgAdmin from here.

Edit the PostgreSQL configuration file `C:\PostgreSQL\data\pg96\postgresql.conf` to make it

listen on 127.0.0.1 only. Adjust the database sizing as necessary. The following example is for 6GB RAM. Do not hasten to overwrite your PgSQL configuration out of hand! When in doubt, use a calculator.

```
listen_addresses = '127.0.0.1'        # what IP address(es) to listen on;
port = 5432             # (change requires restart)
max_connections = 150           # (change requires restart)
superuser_reserved_connections = 3  # (change requires restart)
shared_buffers = 512MB          # min 128kB
work_mem = 12815kB          # min 64kB
maintenance_work_mem = 384MB
dynamic_shared_memory_type = windows    # the default is the first option
wal_level = hot_standby
wal_buffers = 16MB          # min 32kB, -1 sets based on shared_buffers
max_wal_size = 2GB
min_wal_size = 1GB
checkpoint_completion_target = 0.7  # checkpoint target duration, 0.0 - 1.0
max_wal_senders = 5
wal_keep_segments = 32
max_replication_slots = 5
effective_cache_size = 4608MB
default_statistics_target = 100     # range 1-10000
logging_collector = on
log_directory = 'C:/POSTGR~1/data/logs/pg96'
log_filename = 'postgresql-%a.log'
log_truncate_on_rotation = on
log_checkpoints = on
log_line_prefix =  '%t [%p]: [%l-1] user=%u,db=%d,app=%a,client=%h '
log_lock_waits = on
log_temp_files = 0
log_timezone = 'Europe/Belgrade'
update_process_title = off
track_io_timing = on
log_autovacuum_min_duration = 0
datestyle = 'iso, mdy'
timezone = 'Europe/Belgrade'
lc_messages = 'English_United States.1252'            # locale for system
error message
lc_monetary = 'English_United States.1252'            # locale for monetary
formatting
lc_numeric = 'English_United States.1252'         # locale for number
formatting
lc_time = 'English_United States.1252'                # locale for time
formatting
default_text_search_config = 'pg_catalog.english'
```

Configure the authentication in the `C:\PostgreSQL\data\pg96\pg_hba.conf` to accept passwords. The basic configuration file should then look like this:

```
# TYPE  DATABASE        USER            CIDR-ADDRESS            METHOD
```

```
# IPv4 local & remote connections:
host    all             all             127.0.0.1/32            md5
host    all             all             0.0.0.0/0               md5
# IPv6 local connections:
host    all             all             ::1/128                 md5
```

Open Windows services' dialogue (Win+r and type **services.msc** therein). Look for the "PostgreSQL 9.6 Server" service and set its `StartupType` to `Automatic`. Then start the service.

If you install the database onto a server distinct from the one on which the CzechIdM application itself (Tomcat etc.) is mounted, don't forget to configure PostgreSQL to allow remote SSL connection from that server.

Java

Install the Oracle JDK (minimal version is 1.8). You can download it from here. Be sure to download the **JDK**, and not only **JRE**.

Install the Java into a standard directory in Program Files. Having finished the installation, it is necessary to set up `JAVA\_HOME` and PATH variables. Open the **sysdm.cpl** (Win+r ant type sysdm.cpl) dialogue and navigate to `> Advanced > Environment Variables`. Add system-wide variable `JAVA\_HOME=C:\Program Files\Java\jdk1.8.0\_152` (**adjust the path accordingly to the Java version you just installed**). Add the `%JAVA_HOME%\bin` to the PATH. Then run `java -version` from the windows cmd to check if it is working properly.

Tomcat

Download and install the latest 8.5 branch of Apache Tomcat from here. Use the Windows installer.

- Leave the installation paths on default.
- Let the setup create `admin` user for the Tomcat console.
- Modify the JRE `path` to make it point to installe Oracle **JDK**.
- Do not install the example application.
- Let the setup create a Tomcat windows service.

After installation, run the **Monitor Tomcat** application from the Start menu. Configure following settings:

- `initial memory pool:` 512MB (example for about 5GB RAM).
- `maximum memory pool:` 4096MB (example for about 5GB RAM).
- Add `C:\CzechIdM\etc;C:\CzechIdM\lib;C:\CzechIdM\lib\\*;` to the **beginning of the** CLASSPATH.

Configure addresses the server will listen on. Open the `server.xml` configuration file in the Tomcat installation. Make these changes:

- Add `address="127.0.0.1"` to the **8080/tcp** and **8009/tcp** connectors. This will make Tomcat listen only on localhost.
- Change port number `8005` to `-1` at the Shutdown Port setting. This will effectively turn off the shutdown port.

Use the **services.msc** dialogue to set the Apache Tomcat `StartupType` to `Automatic (Delayed Start)`. This will make the application container start after the PostgreSQL database.

`For production use, we strongly advise to remove all Tomcat's management` applications from the container.

- Locate the `webapps` folder in the Tomcat installation and delete everything that is inside.

For roles and advanced management configuration, please see the relevant chapters in the Server Preparation - Linux tutorial.

Apache httpd as a reverse proxy

It is possible to open Apache Tomcat to the network directly, but somewhat inconvenient. You want the users to access CzechIdM on user-friendly ports 80/tcp or 443/tcp. So we use Apache httpd as a reverse proxy and add a few security features along the way. Apache httpd will allow access to data via https on port 443/tcp and http on port 80/tcp. Communication via http protocol is enabled, but we redirect all communication to https. Communication between Apache httpd and Tomcat takes place on local machine via AJP protocol. In httpd, there will be mod_security installed (optional but recommended), which serves as an application firewall.

The configuration example is written for the server which allows access to its services under the name "demo.czechidm.com".

HTTPd installation and configuration

First, install necessary VCredist library.

Download Apache HTTPd from the apachelounge distribution and unpack it into `C:\apache24` folder.

Fire up an elevated shell and install the Apache HTTPd service:

```
cd C:\apache24\bin
httpd.exe -k install
```

Open the **services.msc** and reconfigure "Apache2.4" service to have `StartupType=Automatic (Delayed start)`.

Configure the HTTPd in its core config file `C:\Apache24\conf\httpd.conf`. You can use the

following file, just replace values for `ServerAdmin` and `ServerName`.

httpd.conf

```
ServerRoot "c:/Apache24"

Listen 80

LoadModule access_compat_module modules/mod_access_compat.so
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule allowmethods_module modules/mod_allowmethods.so
LoadModule asis_module modules/mod_asis.so
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule authn_core_module modules/mod_authn_core.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule authz_core_module modules/mod_authz_core.so
LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule authz_host_module modules/mod_authz_host.so
LoadModule authz_user_module modules/mod_authz_user.so
LoadModule autoindex_module modules/mod_autoindex.so
LoadModule cgi_module modules/mod_cgi.so
LoadModule deflate_module modules/mod_deflate.so
LoadModule dir_module modules/mod_dir.so
LoadModule env_module modules/mod_env.so
LoadModule filter_module modules/mod_filter.so
LoadModule headers_module modules/mod_headers.so
LoadModule include_module modules/mod_include.so
LoadModule isapi_module modules/mod_isapi.so
LoadModule log_config_module modules/mod_log_config.so
#LoadModule log_debug_module modules/mod_log_debug.so
LoadModule mime_module modules/mod_mime.so
#LoadModule mime_magic_module modules/mod_mime_magic.so
LoadModule negotiation_module modules/mod_negotiation.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_ajp_module modules/mod_proxy_ajp.so
#LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
#LoadModule proxy_connect_module modules/mod_proxy_connect.so
#LoadModule proxy_express_module modules/mod_proxy_express.so
#LoadModule proxy_html_module modules/mod_proxy_html.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule proxy_wstunnel_module modules/mod_proxy_wstunnel.so
#LoadModule reqtimeout_module modules/mod_reqtimeout.so
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
LoadModule ssl_module modules/mod_ssl.so
LoadModule unique_id_module modules/mod_unique_id.so
#LoadModule vhost_alias_module modules/mod_vhost_alias.so
LoadModule security2_module modules/mod_security2.so
```

```
<IfModule unixd_module>
# jsme na oknech, tohle se nepouzije
    User daemon
    Group daemon
</IfModule>

# 'Main' server configuration
#
ServerAdmin root@demo.czechidm.com
ServerName demo.czechidm.com

<Directory />
    AllowOverride none
    Require all denied
</Directory>

DocumentRoot "c:/Apache24/htdocs"
<Directory "c:/Apache24/htdocs">
    Options -Indexes -FollowSymLinks -MultiViews
    AllowOverride None
    Require all granted
</Directory>

<IfModule dir_module>
    DirectoryIndex index.html
</IfModule>

<Files ".ht*">
    Require all denied
</Files>

ErrorLog "logs/error.log"
LogLevel warn

<IfModule log_config_module>
    LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-
Agent}i\"" combined
    LogFormat "%h %l %u %t \"%r\" %>s %b" common
    <IfModule logio_module>
      # You need to enable mod_logio.c to use %I and %O
      LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-
Agent}i\" %I %O" combinedio
    </IfModule>
    CustomLog "logs/access.log" common
    #CustomLog "logs/access.log" combined
</IfModule>

<IfModule alias_module>
    ScriptAlias /cgi-bin/ "c:/Apache24/cgi-bin/"
</IfModule>
```

```
<IfModule cgid_module>
    #Scriptsock cgisock
</IfModule>

<Directory "c:/Apache24/cgi-bin">
    AllowOverride None
    Options None
    Require all granted
</Directory>

<IfModule headers_module>
    RequestHeader unset Proxy early
</IfModule>

<IfModule mime_module>
    TypesConfig conf/mime.types
    AddType application/x-compress .Z
    AddType application/x-gzip .gz .tgz
</IfModule>

# Virtual hosts
Include conf/extra/httpd-vhosts.conf

# Configure mod_proxy_html to understand HTML4/XHTML1
<IfModule proxy_html_module>
    Include conf/extra/proxy-html.conf
</IfModule>

# Secure (SSL/TLS) connections
Include conf/extra/httpd-ssl.conf

# Note: The following must must be present to support
#       starting without SSL on platforms with no /dev/random
equivalent
#       but a statically compiled-in mod_ssl.
<IfModule ssl_module>
    SSLRandomSeed startup builtin
    SSLRandomSeed connect builtin
</IfModule>

# Include modsec
# if you do not want to use it, comment-out the section below
<IfModule mod_security2.c>
    Include conf/extra/modsec.conf
</IfModule>
```

Configure the HTTP→HTTPS redirect in the `C:\Apache24\conf\extra\httpd-vhosts.conf`. Replace `demo.czechidm.com` with the name of your server:

httpd-vhosts.conf

```
# Virtual Hosts
#
# Required modules: mod_log_config

<VirtualHost *:80>
    ServerName demo.czechidm.com
    ErrorLog "logs/demo.czechidm.com-error.log"
    CustomLog "logs/demo.czechidm.com-access.log" common

# this is for stable deployment
    Redirect permanent / https://demo.czechidm.com

# this one is for debugging before going live
#    Redirect / https://demo.czechidm.com
</VirtualHost>
```

Configure the HTTPS virtual host in the `C:\Apache24\conf\extra\httpd-ssl.conf` file. Change `demo.czechidm.com` to the name of your server.

In some cases older clients (i.e. IE10 and older, Java6, etc.) will not be able to communicate with IdM due to the SSL settings. If this is your case, you may need to slacken the cipher settings a bit.

httpd-ssl.conf

```
Listen 443

SSLCipherSuite
ALL:!ADH:!EXPORT:!SSLv2:RC4+RSA:+HIGH:+MEDIUM:!LOW:!RC4:!3DES+SHA:!IDEA
SSLProxyCipherSuite HIGH:MEDIUM:!MD5:!RC4:!3DES
SSLHonorCipherOrder on
SSLProtocol all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
SSLProxyProtocol all -SSLv2 -SSLv3
SSLPassPhraseDialog  builtin
SSLSessionCache        "shmcb:c:/Apache24/logs/ssl_scache(512000)"
SSLSessionCacheTimeout  300


<VirtualHost _default_:443>
ServerName demo.czechidm.com
ServerAdmin root@demo.czechidm.com
ErrorLog "c:/Apache24/logs/demo.czechidm.com_ssl-error.log"
TransferLog "c:/Apache24/logs/demo.czechidm.com_ssl-access.log"
CustomLog "c:/Apache24/logs/demo.czechidm.com_ssl-request.log" "%t %h
%{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
```

```
SSLEngine on

SSLCertificateFile "c:/Apache24/conf/server.crt"
SSLCertificateKeyFile "c:/Apache24/conf/server.key"
#SSLCertificateChainFile "c:/Apache24/conf/server-chain.crt"

SSLVerifyClient none

<FilesMatch "\.(cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
</FilesMatch>
<Directory "c:/Apache24/cgi-bin">
    SSLOptions +StdEnvVars
</Directory>

BrowserMatch "MSIE [2-5]" nokeepalive ssl-unclean-shutdown
downgrade-1.0 force-response-1.0

# workaround for bad font handling in IE 11
<LocationMatch "/idm/.*(\.ttf|\.woff2|\.eot)$">
        Header set Cache-Control "no-cache, public, must-revalidate,
proxy-revalidate"
</LocationMatch>

ProxyRequests     off
ProxyPreserveHost on
ProxyAddHeaders on
ProxyPass / ajp://127.0.0.1:8009/
ProxyPassReverse / ajp://127.0.0.1:8009/

RewriteEngine On
RewriteRule "^/$"  "/idm/" [R]

<IfModule mod_security2.c>
        SecRuleRemoveById 981173
        SecRuleRemoveById 960015
        SecRuleRemoveById 950109

        # Allow Czech signs
        SecRuleRemoveById 981318
        SecRuleRemoveById 981242
        SecRuleRemoveById 960024
        SecRuleRemoveById 981245

        # Too restrictive for login format
        SecRuleRemoveById 960035

        # Needed by Websockets
        <Location "/idm/api/v1/websocket-info/">
                SecRuleRemoveById 970901
        </Location>
```

```
        # These break Certificate Authority module
        <Location "/idm/api/v1/crt/certificates/action/validate">
                SecRuleRemoveById 960915
                SecRuleRemoveById 200003
        </Location>

    # Modsec can throw false positives on some files due to multipart
boundary check
    <Location "/idm/api/v1/attachments/upload">
        SecRuleRemoveById 960915
        SecRuleRemoveById 200003
    </Location>

        # do not log request/response body
        SecAuditLogParts ABFHZ
</IfModule>

<IfModule mod_deflate.c>
        # Compress HTML, CSS, JavaScript, Text, XML and fonts
        AddOutputFilterByType DEFLATE application/javascript
        AddOutputFilterByType DEFLATE application/rss+xml
        AddOutputFilterByType DEFLATE application/vnd.ms-fontobject
        AddOutputFilterByType DEFLATE application/x-font
        AddOutputFilterByType DEFLATE application/x-font-opentype
        AddOutputFilterByType DEFLATE application/x-font-otf
        AddOutputFilterByType DEFLATE application/x-font-truetype
        AddOutputFilterByType DEFLATE application/x-font-ttf
        AddOutputFilterByType DEFLATE application/x-javascript
        AddOutputFilterByType DEFLATE application/xhtml+xml
        AddOutputFilterByType DEFLATE application/xml
        AddOutputFilterByType DEFLATE font/opentype
        AddOutputFilterByType DEFLATE font/otf
        AddOutputFilterByType DEFLATE font/ttf
        AddOutputFilterByType DEFLATE image/svg+xml
        AddOutputFilterByType DEFLATE image/x-icon
        AddOutputFilterByType DEFLATE text/css
        AddOutputFilterByType DEFLATE text/html
        AddOutputFilterByType DEFLATE text/javascript
        AddOutputFilterByType DEFLATE text/plain
        AddOutputFilterByType DEFLATE text/xml
        AddOutputFilterByType DEFLATE application/json
        AddOutputFilterByType DEFLATE application/hal+json

        # Remove browser bugs (only needed for really old browsers)
        BrowserMatch ^Mozilla/4 gzip-only-text/html
        BrowserMatch ^Mozilla/4\.0[678] no-gzip
        BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
        Header append Vary User-Agent
</IfModule>
```

```
</VirtualHost>
```

Supply SSL certificate and key in x509 PEM form to `c:/Apache24/conf/server.key` and `c:/Apache24/conf/server.crt` files. Apache HTTPd will not start without those files. If you need to generate some ad-hoc certificates, use for example this guide. You can easily invoke the **openssl** tool from the Git Bash prompt.

mod_security installation

Download the mod_security module from the Apache Lounge project. Unpack the zip and perform following actions:

- Copy the `mod\_security2.so` int `C:\Apache24\modules` directory.
- Copy `libcurl.dll` and `yajl.dll` into `C:\Apache24\bin` directory.

Create general mod_security configuration file `C:\Apache24\conf\extra\modsec.conf`:

modsec.conf

```
<IfModule mod_security2.c>
    # ModSecurity Core Rules Set configuration
    IncludeOptional conf/modsecurity_win.d/*.conf
    IncludeOptional conf/modsecurity_win.d/activated_rules/*.conf

    # Default recommended configuration
    SecRuleEngine On
    SecRequestBodyAccess On
    SecRule REQUEST_HEADERS:Content-Type "text/xml" \
"id:'200000',phase:1,t:none,t:lowercase,pass,nolog,ctl:requestBodyProce
ssor=XML"
    SecRequestBodyLimit 13107200
    SecRequestBodyNoFilesLimit 131072
    SecRequestBodyInMemoryLimit 131072
    SecRequestBodyLimitAction Reject
    SecRule REQBODY_ERROR "!@eq 0" \
    "id:'200001', phase:2,t:none,log,deny,status:400,msg:'Failed to
parse request body.',logdata:'%{reqbody_error_msg}',severity:2"
    SecRule MULTIPART_STRICT_ERROR "!@eq 0" \
    "id:'200002',phase:2,t:none,log,deny,status:44,msg:'Multipart
request body \
    failed strict validation: \
    PE %{REQBODY_PROCESSOR_ERROR}, \
    BQ %{MULTIPART_BOUNDARY_QUOTED}, \
    BW %{MULTIPART_BOUNDARY_WHITESPACE}, \
    DB %{MULTIPART_DATA_BEFORE}, \
    DA %{MULTIPART_DATA_AFTER}, \
    HF %{MULTIPART_HEADER_FOLDING}, \
```

```
    LF %{MULTIPART_LF_LINE}, \
    SM %{MULTIPART_MISSING_SEMICOLON}, \
    IQ %{MULTIPART_INVALID_QUOTING}, \
    IP %{MULTIPART_INVALID_PART}, \
    IH %{MULTIPART_INVALID_HEADER_FOLDING}, \
    FL %{MULTIPART_FILE_LIMIT_EXCEEDED}'"

    SecRule MULTIPART_UNMATCHED_BOUNDARY "!@eq 0" \
    "id:'200003',phase:2,t:none,log,deny,status:44,msg:'Multipart
parser detected a possible unmatched boundary.'"

    SecPcreMatchLimit 1000
    SecPcreMatchLimitRecursion 1000

    SecRule TX:/^MSC_/ "!@streq 0" \
            "id:'200004',phase:2,t:none,deny,msg:'ModSecurity internal
error flagged: %{MATCHED_VAR_NAME}'"

    SecResponseBodyAccess Off
#    SecDebugLog /var/log/httpd/modsec_debug.log
#    SecDebugLogLevel 0
    SecAuditEngine RelevantOnly
    SecAuditLogRelevantStatus "^(?:5|4(?!04))"
    SecAuditLogParts ABIJDEFHZ
    SecAuditLogType Serial
    SecAuditLog logs/modsec_audit.log
    SecArgumentSeparator &
    SecCookieFormat 0
    SecTmpDir modsec_tmp
    SecDataDir modsec_lib
</IfModule>
```

Create empty directories `C:\Apache24\modsec\_tmp` and `C:\Apache24\modsec\_lib` for mod_security working data.

Mod_security will become operational but will have no filtering rules. To obtain filtering rules, please visit Mod Security project homepage. **Remember to obtain 2.x version of rules, not the newest 3.x version!**

For commercial deployment of CzechIdM, we have prepared a pack of mod_security rules which you need to just unpack into `C:\Apache24\conf` directory, where it creates a `modsecurity\_win.d` folder full of rules.

Mod Security rules package

(login required).

Now you can start the Apache HTTPd using its service. If it fails to start, check the Windows EventLog for errors.