

# Adding authentication to room creation

You probably want to restrict the creation of new rooms to some users, but still allow everyone with the link to join a room. For this we can configure `jicofo`:

First edit the `/etc/prosody/conf.avail/jitsi.yourdomain.example.cfg.lua`. In the main `VirtualHost`, replace `anonymous` authentication with `internal_plain`. Next up create a new `VirtualHost` below for e.g. `guest.jitsi.yourdomain.example` with `anonymous` authentication.

Afterwards the file should look similar to this:

```
VirtualHost "jitsi.yourdomain.example"      -- enabled = false -- Remove this line to
enable this host

    authentication = "internal_plain"        -- Properties below are modified by jitsi-
meet-tokens package config
    -- and authentication above is switched to "token"      --
app_id="example_app_id"
    --app_secret="example_app_secret"        -- Assign this host a certificate for TLS,
otherwise it would use the one
    -- set in the global section (if any).                -- Note that old-style SSL on port 5223
only supports one certificate, and will always
    -- use the global one.
    ssl = {          key =
"/etc/prosody/certs/jitsi.yourdomain.example.key";          certificate =
"/etc/prosody/certs/jitsi.yourdomain.example.crt";
    }
    -- we need bosh
modules_enabled = {
    "bosh";
    "pubsub";
    "ping"; -- Enable mod_ping
}
c2s_require_encryption = false
VirtualHost "guest.jitsi.yourdomain.example"
```

```
authentication = "anonymous"
c2s_require_encryption = false
```

Next up, you have to edit the `config.js` at `/etc/jitsi/meet/jitsi.yourdomain.example-config.js`.

It should already have a line for `anonymousdomain` which you only need to uncomment and specify the domain of the new `VirtualHost`.

```
var config = {
  hosts: {
    // XMPP domain.
    domain: 'jitsi.yourdomain.example',
    // When using authentication, domain for guest users.      anonymousdomain:
    'guest.jitsi.yourdomain.example',
    [...]
  }
}
```

Now we need to enable the authentication in `jicofo`. For this we create `/etc/jitsi/jicofo/sip-communicator.properties` and set it to

```
org.jitsi.jicofo.auth.URL=XMPP:jitsi.yourdomain.example
```

After every config is set, we can restart `jicofo` and `prosody`

```
sudo systemctl restart prosody.service
sudo systemctl restart jicofo.service
```

And lastly we can create users using `prosodyctl`:

```
prosodyctl register <username> jitsi-meet.example.com <password>
```

Now when you create a new room, you can click on `I am the host` and enter the username/password. Every following user will be able to join directly.

## Waiting for the host ...

The conference **test-room** has not yet started. If you are the host then please authenticate. Otherwise, please wait for the host to arrive.

I am the host

# Using existing user databases

As jitsi is using Prosody for this, you can use e.g. [IMAP](#), [LDAP](#), [Wordpress](#) and [many more](#) for authentication.

But if you just want to limit the instance to the people you know, a single shared user account as set up above might be enough.

---

Revision #8

Created Mon, Mar 16, 2020 2:53 PM by [Lukas](#)

Updated Mon, Mar 30, 2020 11:52 AM by [Lukas](#)