



Skydel HTML Server Manual

– Revision 1.1

Thomas Topp

Oct. 2021

Contents

1	Introduction	4
2	Usage	4
2.1	Browser	4
2.2	Xpra over SSH	4
3	Setup	5
3.1	Ubuntu Server 20.04 LTS	6
3.1.1	Users and Groups	6
3.1.2	Disable Hibernate	6
3.1.3	Nvidia driver	6
3.2	Skydel Software	7
3.2.1	Installing skydel-sdx	7
3.2.2	Skydel-SDX folder linkage	7
3.2.3	OpenStreetMap SSL Error	7
3.3	HTML Server	8
3.3.1	Certbot	8
3.3.2	Nginx	8
3.3.3	Xpra	9
4	Access outputs	10
5	Update	10
	Appendices	11
A	Linux User permissions	11
B	Nginx configuration files	12
B.1	/etc/nginx/nginx.conf	12
B.2	/etc/nginx/snippets/proxy-params.conf	13
B.3	/etc/nginx/snippets/<address>.cert.conf	13
B.4	/etc/nginx/snippets/ssl-params.conf	14
B.5	/etc/nginx/sites-available/xpra	14
C	Xpra configuration files	15
C.1	15
C.2	/usr/lib/systemd/system/xpra.service	16
C.3	/etc/xpra/conf.d/05_features.conf	17
C.4	/etc/xpra/conf.d/10_network.conf	19
C.5	/etc/xpra/conf.d/12_ssl.conf	21
C.6	/etc/xpra/conf.d/15_file_transfers.conf	23
C.7	/etc/xpra/conf.d/16_printing.conf	24
C.8	/etc/xpra/conf.d/20_sound.conf	25
C.9	/etc/xpra/conf.d/30_picture.conf	26
C.10	/etc/xpra/conf.d/35_webcam.conf	28
C.11	/etc/xpra/conf.d/40_client.conf	29
C.12	/etc/xpra/conf.d/42_client_keyboard.conf	31

C.13	/etc/xpra/conf.d/50_server_network.conf	33
C.14	/etc/xpra/conf.d/55_server_x11.conf	36
C.15	/etc/xpra/conf.d/60_server.conf	38
C.16	/etc/xpra/conf.d/65_proxy.conf	42

1 Introduction

This technical note explains the basic steps in order to set up an Ubuntu 20.04 Server for the Skydel Simulation Engine¹ from Orolia². If you follow the installation steps and have Xpra³ running on the server you should be able remotely access your Skydel Simulation Engine over SSH and browser.

2 Usage

2.1 Browser

After setting up the server just visit the URL of your server. There you will see a login screen (figure 1) where you have to provide the username and password which was set while setting up the server. Afterwards the Skydel Simulation Engine will start as a window inside the browser. This method does not need to install any dependencies on the client side except a browser.

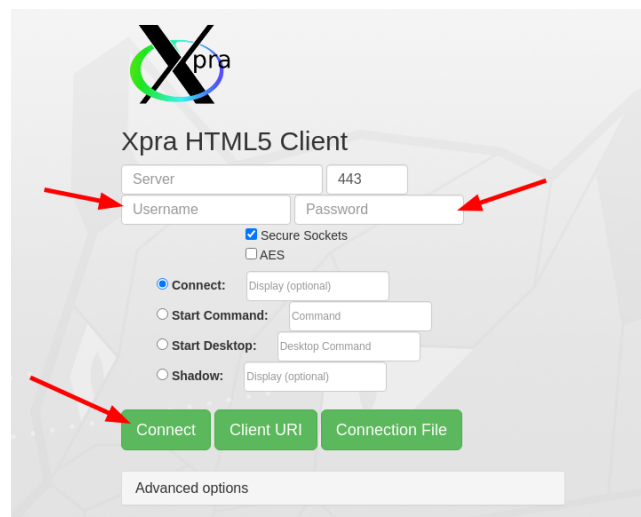


Figure 1: Xpra login screen

2.2 Xpra over SSH

With this method the Skydel Simulation Engine can be forwarded through the SSH Protocol. This way the simulator will run on the server but the Graphical User Interface (GUI) will be forwarded directly to the client computer and can be used like any other program in its own window.

```
$ xpra start ssh:<username>@<address> --dpi=100
```

For this command the user will need to install the xpra software⁴ on the client computer. The command will ask the user to type the password and then the Skydel Simulation Engine window should open. Please refer to the xpra wiki⁵ for a detailed description of the commands.

¹<https://www.orolia.com/products/gnss-simulation/skydel>

²<https://www.orolia.com/>

³<https://xpra.org/>

⁴<https://github.com/Xpra-org/xpra/wiki/Download>

⁵<https://github.com/Xpra-org/xpra/tree/master/docs/Usage>

3 Setup

In this section the general installation process of the Ubuntu server is described as well as the html server setup.

For the manual it is assumed that there should be three different linux users. This of course can be extended to more users or only a single user can be used.

- **simulator**: Used to run the Skydel software, not used for logins
- **staff**: For trusted employees
- **external**: For external users

The **simulator** user is needed because all instances of the Skydel software have to run as the same linux user. So the idea is to login as **staff** or **external** user but to run the software as **simulator** user. That way external users don't have access on internal research files.

3.1 Ubuntu Server 20.04 LTS

Please follow the instructions on ubuntu.com to install a basic Ubuntu server.

3.1.1 Users and Groups

Create the linux users

```
$ sudo adduser simulator
$ sudo adduser staff
$ sudo adduser external
```

Alter the permissions to the home directories of the users. For an explanation of the linux permissions see Appendix A

```
$ sudo chmod 770 /home/staff/ /home/external/
$ sudo chmod 777 /home/simulator/
$ sudo -u staff mkdir -p /home/staff/Documents
$ sudo -u external mkdir -p /home/external/Documents
$ sudo chmod 775 /home/staff/Documents /home/external/Documents
```

Add the users `staff` and `external` to the `simulator` group

```
$ sudo usermod -aG simulator external
$ sudo usermod -aG simulator staff
```

Allow the users `staff` and `external` to run the Skydel software as user `simulator`

```
$ sudo visudo
# Allow these users to execute the skydel-sdx command without
# password prompt
# xhost + && sudo -u simulator -g staff -i skydel-sdx
staff ALL=(simulator:staff) NOPASSWD: /bin/bash -c skydel-sdx
external ALL=(simulator:external) NOPASSWD: /bin/bash -c skydel-sdx
```

3.1.2 Disable Hibernate

Make sure the server is not hibernating as it can be a hassle to remotely turn it on again

```
$ sudo systemctl mask sleep.target suspend.target hibernate.target
hybrid-sleep.target
```

3.1.3 Nvidia driver

Install a recent driver version for your Nvidia graphics card. This can be for example done like this

```
$ sudo apt install nvidia-460 nvidia-settings
```

3.2 Skydel Software

3.2.1 Installing skydel-sdx

Download the newest software from <https://users.skydelsolutions.com/> and install it

```
$ sudo dpkg -i ~/Downloads/skydel-sdx.deb
```

Ubuntu 20.04 comes with a different libm version, so we need to preload

```
$ sudo bash -c 'echo "export LD_PRELOAD=/lib/x86_64-linux-gnu/libm.so
.6" >> /home/simulator/.profile
```

When first launching the simulator software or when you want to update the license over xpra you could get an error that the license file could not be opened. If this happens launch the software physically on the server with monitor and keyboard. Afterwards it can be started over xpra with the new license.

3.2.2 Skydel-SDX folder linkage

To access the output files, we can link the Skydel-SDX folder to the different users. In this example the `staff` user has access to the full folder, while the `external` user only has access to the Output and API folders.

```
$ sudo ln -s /home/simulator/Documents/Skydel-SDX/ /home/staff/
Documents/
$ sudo -u external mkdir /home/external/Documents/Skydel-SDX
$ sudo ln -s /home/simulator/Documents/Skydel-SDX/Output /home/external
/Documents/Skydel-SDX
$ sudo ln -s /home/simulator/Documents/Skydel-SDX/API /home/external/
Documents/Skydel-SDX
$ sudo chmod -R 755 /home/simulator/Documents/Skydel-SDX/API
```

3.2.3 OpenStreetMap SSL Error

Running the Skydel software under Ubuntu 20.04 you will get a SSL Error. For details please see the [learn.orolia Forum](https://learn.orolia.com/t/openstreetmap-ssl-error/148)⁶. To fix this, simply install the appropriate OpenSSL version (OpenSSL 1.0.2u)

```
$ sudo apt install build-essential checkinstall zlib1g-dev -y
$ curl https://www.openssl.org/source/old/1.0.2/openssl-1.0.2u.tar.gz
$ tar -xf openssl-1.0.2u.tar.gz
$ cd openssl-1.0.2u
$ ./config --prefix=/usr/local/ssl --openssldir=/usr/local/ssl shared
zlib
$ sudo make
$ sudo make test
$ sudo make install
$ sudo bash -c 'echo "/usr/local/ssl/lib" > /etc/ld.so.conf.d/openssl
-1.0.2u.conf'
$ sudo ldconfig -v
$ cd ..
$ rm -r openssl-1.0.2u
```

⁶<https://learn.orolia.com/t/openstreetmap-ssl-error/148>

3.3 HTML Server

3.3.1 Certbot

Certbot enables you to get a certificate, which is needed to enable HTTPS access.

Install the nginx web server

```
$ sudo apt install nginx
```

Install Certbot

```
$ sudo snap install core; sudo snap refresh core
$ sudo snap install --classic certbot
```

Link the executable if this is not done by the snap already

```
$ sudo ln -s /snap/bin/certbot /usr/bin/certbot
```

Get a certificate

```
$ sudo certbot --nginx

# Test automatic renewal
$ sudo certbot renew --dry-run
```

Create a cronjob if it wasn't done automatically

```
$ sudo crontab -e
# minute (0-59),
# |      hour (0-23),
# |      |      day of the month (1-31),
# |      |      |      month of the year (1-12),
# |      |      |      |      day of the week (0-6 with 0=Sunday).
# |      |      |      |      |      commands
0    0,12    *    *    *    /usr/bin/python3 -c 'import
    random; import time; time.sleep(random.random() * 3600)' && /usr
    /bin/certbot renew --quiet --deploy-hook "/usr/sbin/service
    nginx reload"
```

3.3.2 Nginx

Use the configuration files provided in Appendix B and make sure to replace all placeholders.

- /etc/nginx/nginx.conf
- /etc/nginx/snippets/proxy-params.conf
- /etc/nginx/snippets/<address>.cert.conf
- /etc/nginx/snippets/ssl-params.conf
- /etc/nginx/sites-available/xpra

Link the xpra server file and remove the default server file

```
$ sudo ln -s /etc/nginx/sites-available/xpra /etc/nginx/sites-enabled/
$ sudo rm /etc/nginx/sites-enabled/default
```

Start and enable the nginx server

```
$ sudo systemctl enable nginx
$ sudo systemctl start nginx
```


3.3.3 Xpra

The general installation instructions can be found at the xpra GitHub⁷. Here are the commands to install it on ubuntu.

```
# install https support for apt (which may be installed already):
$ sudo apt update
$ sudo apt install apt-transport-https
# add Xpra GPG key
$ wget -q https://xpra.org/gpg.asc -O- | sudo apt-key add -
# add Xpra repository
$ sudo add-apt-repository "deb https://xpra.org/ focal main"
# install Xpra package
$ sudo apt update
$ sudo apt install xpra
```

Xpra has different authentication methods⁸. Here the authentication with a sqlite database is explained.

The sqlite users will be used for authenticating in the browser. They in general could have different names and passwords than the linux users. But for convenience the same names are chosen here.

```
$ sudo chown root:root /etc/xpra/xpra-auth.sdb
$ sudo chmod 660 /etc/xpra/xpra-auth.sdb
$ sudo python3 /usr/lib/python3/dist-packages/xpra/server/auth/
  sqlite_auth.py /etc/xpra/xpra-auth.sdb list
$ sudo python3 /usr/lib/python3/dist-packages/xpra/server/auth/
  sqlite_auth.py /etc/xpra/xpra-auth.sdb add staff <password> staff
  staff
$ sudo python3 /usr/lib/python3/dist-packages/xpra/server/auth/
  sqlite_auth.py /etc/xpra/xpra-auth.sdb add external <password>
  external external
```

Add the users to the xpra group (needed to create sockets)

```
$ sudo gpasswd -a simulator xpra
$ sudo gpasswd -a external xpra
$ sudo gpasswd -a staff xpra
```

Use the configuration files provided in Appendix C and make sure to replace all placeholders.

- /etc/default/xpra
- /usr/lib/systemd/system/xpra.service
- /etc/xpra/conf.d/05_features.conf
- /etc/xpra/conf.d/10_network.conf
- /etc/xpra/conf.d/12_ssl.conf
- /etc/xpra/conf.d/15_file_transfers.conf
- /etc/xpra/conf.d/16_printing.conf
- /etc/xpra/conf.d/20_sound.conf

⁷<https://github.com/Xpra-org/xpra/wiki/Download>

⁸<https://github.com/Xpra-org/xpra/blob/master/docs/Usage/Authentication.md>

- /etc/xpra/conf.d/30_picture.conf
- /etc/xpra/conf.d/35_webcam.conf
- /etc/xpra/conf.d/40_client.conf
- /etc/xpra/conf.d/42_client_keyboard.conf
- /etc/xpra/conf.d/50_server_network.conf
- /etc/xpra/conf.d/55_server_x11.conf
- /etc/xpra/conf.d/60_server.conf
- /etc/xpra/conf.d/65_proxy.conf

Start and enable the xpra server

```
$ sudo systemctl enable xpra
$ sudo systemctl start xpra
```

4 Access outputs

In order to access the simulator outputs we need to provide remote access to the local files on the server. A very easy solution is to mount the server files over SFTP as external storage into a cloud service.

5 Update

Updating the xpra server happens with the system update commands. However the xpra configuration files could be overridden, so make sure to check them after a system update.

```
$ sudo apt update
$ sudo apt upgrade
```

Appendices

A Linux User permissions

Every file in Unix has the following attributes⁹

- **Owner permissions** - The owner's permissions determine what actions the owner of the file can perform on the file.
- **Group permissions** - The group's permissions determine what actions a user, who is a member of the group that a file belongs to, can perform on the file.
- **Other (world) permissions** - The permissions for others indicate what action all other users can perform on the file.

The file access modes are as follows

- **Read** - Grants the capability to read, i.e., view the contents of the file.
- **Write** - Grants the capability to modify, or remove the content of the file.
- **Execute** - User with execute permissions can run a file as a program.

The following table explains the number format to set permissions

Number	Octal Permission Representation	Ref
0	No permission	---
1	Execute permission	--x
2	Write permission	-w-
3	Execute and write permission: 1 (execute) + 2 (write) = 3	-wx
4	Read permission	r--
5	Read and execute permission: 4 (read) + 1 (execute) = 5	r-x
6	Read and write permission: 4 (read) + 2 (write) = 6	rw-
7	All permissions: 4 (read) + 2 (write) + 1 (execute) = 7	rwx

⁹<https://www.tutorialspoint.com/unix/unix-file-permission.htm>

B Nginx configuration files

Make sure to replace the following placeholders:

- `<address>` - URL (without http/https) where to reach the server
- `<resolver>` - replace with the IP address of your resolver

The config files are embedded into the pdf and can be downloaded by clicking on the title of the box (may not work with all pdf readers)

B.1 `/etc/nginx/nginx.conf`

 `/etc/nginx/nginx.conf`

```
user www-data;
worker_processes auto;
pid /run/nginx.pid;
include /etc/nginx/modules-enabled/*.conf;

events {
    worker_connections 768;
    # multi_accept on;
}

http {
    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 2048;
    server_tokens off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    gzip on;

    server {
        listen 80;
        #listen [::]:80;
        server_name <address>;
        # enforce https
        return 301 https://$server_name$request_uri;
    }

    # Virtual Host Configs
    include /etc/nginx/conf.d/*.conf;
    include /etc/nginx/sites-enabled/*;
}
```

B.2 /etc/nginx/snippets/proxy-params.conf

 /etc/nginx/snippets/proxy-params.conf

```
proxy_set_header Host $host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto $scheme;
proxy_set_header X-Forwarded-Host $server_name;
proxy_set_header X-Forwarded-Ssl on;
proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection "upgrade";
proxy_http_version 1.1;
```

B.3 /etc/nginx/snippets/<address>.cert.conf

 /etc/nginx/snippets/<address>.cert.conf

```
# certs sent to the client in SERVER HELLO are concatenated in
    ssl_certificate
ssl_certificate /etc/letsencrypt/live/<address>/fullchain.pem; #
    managed by Certbot
ssl_certificate_key /etc/letsencrypt/live/<address>/privkey.pem; #
    managed by Certbot

# verify chain of trust of OCSP response using Root CA and
    Intermediate certs
ssl_trusted_certificate /etc/letsencrypt/live/<address>/chain.pem;
```

B.4 /etc/nginx/snippets/ssl-params.conf

 /etc/nginx/snippets/ssl-params.conf

```
# Certbot security settings
include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot

# Diffie-Hellman keys to prevent man-in-the-middle when private server
# key is leaked
ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot

# HSTS (ngx_http_headers_module is required) (63072000 seconds)
add_header Strict-Transport-Security "max-age=63072000" always;

# OCSP stapling
ssl_stapling on;
ssl_stapling_verify on;

# replace with the IP address of your resolver
resolver <resolver>;
```

B.5 /etc/nginx/sites-available/xpra

 /etc/nginx/sites-available/xpra

```
server {
    listen 443 ssl http2;
    #listen      [::]:443 ssl http2; # Ipv6 is disabled on the server
    server_name  <address>;

    access_log /var/log/nginx/skydel-simulator.access.log;
    error_log /var/log/nginx/skydel-simulator.error.log;

    include snippets/<address>.cert.conf;
    include snippets/ssl-params.conf;

    location / {
        include snippets/proxy-params.conf;
        proxy_pass http://127.0.0.1:10000/;
    }
}
```

C Xpra configuration files

Make sure to replace the following placeholders:

- `<address>` - URL (without http/https) where to reach the server

The config files are embedded into the pdf and can be downloaded by clicking on the title of the box (may not work with all pdf readers)

C.1

 `/etc/default/xpra`

```
BIND_TCP=0.0.0.0:10000
#BIND_TCP=0.0.0.0:80
TCP_AUTH=sqlite:filename=/etc/xpra/xpra-auth.sdb
BIND=none
AUTH=none
HTML=on
#DEBUG=auth,proxy,util,x11
DEBUG=
#skip datetime prefix since the log goes to a system logger:
XPRA_LOG_FORMAT="%(message)s"

#permissions and group ownership of /run/xpra
XPRA_SOCKET_DIR_MODE=775
XPRA_SOCKET_DIR_GROUP=xpra
```

C.2 /usr/lib/systemd/system/xpra.service

 /usr/lib/systemd/system/xpra.service

```
[Unit]
Description=Xpra System Server
Wants=avahi-daemon.socket
Documentation=https://xpra.org/trac/wiki/Service man:xpra
After=network.target xpra.socket
Requires=xpra.socket

[Service]
Type=simple
EnvironmentFile=-/etc/default/xpra
ExecStart=/usr/bin/xpra proxy :14500 --daemon=no \
  --bind-tcp=${BIND_TCP} --tcp-auth=${TCP_AUTH} \
  --socket-dirs=/run/xpra --socket-permissions=666 \
  --log-dir=/var/log --pidfile=/run/xpra/proxy.pid --debug=${DEBUG} \
  \
  --html=${HTML}
#   --bind=${BIND} --auth=${AUTH} \
#   --ssl-cert=/etc/xpra/ssl-cert.pem --ssl=on \
#rely on SIGKILL which returns 128+15=143
SuccessExitStatus=0 143
Restart=on-abnormal
PIDFile=/run/xpra/proxy.pid
ProtectSystem=strict
ReadWritePaths=/run/xpra /tmp
#PrivateDevices=true
ProtectKernelTunables=true
ProtectControlGroups=true

[Install]
WantedBy=multi-user.target
```


C.3 /etc/xpra/conf.d/05_features.conf

 /etc/xpra/conf.d/05_features.conf

```
#####
# General Options

# Enable clipboard forwarding:
#clipboard = yes
#clipboard = no
# Can also be used to specify a different keyboard implementation:
#clipboard = GDK
#clipboard = translated
#clipboard = default
#clipboard = auto
clipboard = yes

# Direction of clipboard transfers:
#clipboard-direction = to-server
#clipboard-direction = to-client
clipboard-direction = both

# Forward notifications:
notifications = yes

# Show Xpra's tray menu
tray = yes

# Forward system tray icons:
system-tray = yes

# Debugging:
#debug =
#debug = keyboard,clipboard,tray

# Send ping packets:
# pings = no
# send every second:
# pings = 1
pings = 5

# Allow the client to forward its log output to the server:
#remote-logging = both
#remote-logging = allow
#remote-logging = send
#remote-logging = receive
#remote-logging = no
remote-logging = both
```

```
# Extra environment variables to set:  
# env = XPRA_USE_ALIASES=0  
# env = XPRA_DEFAULT_VFB_RESOLUTION=1920x1080  
# env = XPRA_DEFAULT_DESKTOP_VFB_RESOLUTION=1024x768
```

C.4 /etc/xpra/conf.d/10_network.conf

 /etc/xpra/conf.d/10_network.conf

```
#####
# Network Connection

# Enable shared memory transfers:
mmap = yes

# Set group ownership of the mmap file:
#leave it as it is:
#mmap-group = no
#use a specific group:
#mmap-group = xpra
#use the server socket's group:
#mmap-group = SOCKET
#use the "xpra" group if possible, or fallback to the socket's group:
mmap-group = auto

# The file permissions set on the server's unix domain socket
# (ignored when mmap-group is enabled)
#socket-permissions = 660
socket-permissions = 600

# Share session with other users:
#sharing = yes
#sharing = no
# Leave it up to the client, where it defaults to no
# and this setting can then be changed from the system tray menu:
sharing = no

# Allow sessions to be locked:
#lock = no
#lock = yes
# Leave it up to the client, where it defaults to no
# and this setting can then be changed from the system tray menu:
lock = auto

# Compressors:
#compressors = all
#compressors = none
#compressors = zlib
compressors = lz4, lzo, zlib, brotli

# Default compression (0 to 9):
#compression_level = 1
compression_level = 6

# Packet encoders (at least one is required):
```

```
#packet-encoders = bencode
#packet-encoders = all
packet-encoders = reencode, bencode, yaml

# Socket directories (may be specified more than once):
#socket-dirs = /tmp
#socket-dirs = ~/.xpra
#socket-dirs = /run/xpra
socket-dirs = $XDG_RUNTIME_DIR/xpra
socket-dirs = /run/xpra
socket-dirs = ~/.xpra

# Where to create new sockets
# (otherwise the first "socket-dirs" is used)
#socket-dir = /tmp
#socket-dir = ~/.xpra

# Session idle timeout in seconds:
#idle-timeout = 120
idle-timeout = 0

# Server idle timeout in seconds:
#server-idle-timeout = 600
server-idle-timeout = 0

# Bandwidth limit:
#no limit:
#bandwidth-limit = 0
#1Mbps:
#bandwidth-limit = 1000000
#bandwidth-limit = 1000Kbps
#bandwidth-limit = 1M
#10Mbps:
#bandwidth-limit = 10Mbps
bandwidth-limit = auto
```

C.5 /etc/xpra/conf.d/12_ssl.conf

 /etc/xpra/conf.d/12_ssl.conf

```
#####
# SSL Options
#
# Please refer to the python ssl module for details

# Key file to use:
#ssl-key = /path/to/keyfile
ssl-key = /etc/letsencrypt/live/<address>/privkey.pem

# Certificate file to use:
#ssl-cert = /path/to/cert
ssl-cert = /etc/letsencrypt/live/<address>/fullchain.pem

# Specifies which version of the SSL protocol to use:
#ssl-protocol = SSLv23
ssl-protocol = TLSv1_2

# The ca_certs file contains a set of concatenated 'certification
  authority' certificates:
#ssl-ca-certs = default
#ssl-ca-certs = /path/to/cacertfile
#ssl-ca-certs = /path/to/cacertsdir/

# Whether to try to verify the client's certificates
# and how to behave if verification fails:
#ssl-client-verify-mode = none
#ssl-client-verify-mode = optional
#ssl-client-verify-mode = required
ssl-client-verify-mode = optional

# Whether to try to verify the server's certificates
# and how to behave if verification fails:
#ssl-server-verify-mode = none
#ssl-server-verify-mode = optional
#ssl-server-verify-mode = required
ssl-server-verify-mode = required

# Whether to match the peer cert's hostname:
ssl-check-hostname = on

# Server hostname to check for:
#ssl-server-hostname = localhost

# The following options require Python 2.7.9 or later:
```

```
# The flags for certificate verification operations:
#ssl-verify-flags = DEFAULT
#ssl-verify-flags = CRL_CHECK_LEAF
#ssl-verify-flags = CHECK_CHAIN
ssl-verify-flags = X509_STRICT

# Sets the available ciphers for this SSL object:
#ssl-ciphers = ALL
#ssl-ciphers = HIGH
ssl-ciphers = DEFAULT

# Set of SSL options enabled on this context:
#ssl-options = ALL
#ssl-options = OP_NO_TLSv1,NO_COMPRESSION
ssl-options = ALL,NO_COMPRESSION
```

C.6 /etc/xpra/conf.d/15_file_transfers.conf

 /etc/xpra/conf.d/15_file_transfers.conf

```
#####  
# File transfers  
  
# Receive files  
#file-transfer = ask  
file-transfer = no  
#file-transfer = auto  
  
# Open URL  
open-url = no  
#open-url = auto  
  
# Location where files are received:  
#download-path = ~/Downloads  
#download-path = ~/Desktop  
#download-path = /tmp  
  
# File size limit  
file-size-limit = 100M  
  
# How to open files:  
#open-command = /usr/bin/xdg-open  
  
# Open files  
open-files = no  
#open-files = yes  
#open-files = auto
```

C.7 /etc/xpra/conf.d/16_printing.conf

 /etc/xpra/conf.d/16_printing.conf

```
#####  
# Printer forwarding  
  
# Print support:  
#printing = yes  
printing = no  
  
# To manage printers:  
#lpadmin = lpadmin  
#lpadmin = sudo lpadmin  
lpadmin = /usr/sbin/lpadmin  
  
# Add printer options  
add-printer-options = -E  
add-printer-options = -o printer-is-shared=false  
add-printer-options = -u allow:$USER  
  
# To query printer definitions:  
#lpinfo = lpinfo  
#lpinfo = sudo lpinfo  
lpinfo = /usr/sbin/lpinfo  
  
# To save startup time, use pre-defined printers:  
# Postscript:  
# postscript-printer = drv:///sample.drv/generic.ppd  
postscript-printer = drv:///sample.drv/generic.ppd  
# pdf-printer = /usr/share/ppd/cupsfilters/Generic-PDF_Printer-PDF.ppd  
pdf-printer = /usr/share/ppd/cupsfilters/Generic-PDF_Printer-PDF.ppd
```


C.8 /etc/xpra/conf.d/20_sound.conf

 /etc/xpra/conf.d/20_sound.conf

```
#####  
# Sound  
  
# Forward sound output to clients:  
#speaker = off  
#speaker = disabled  
speaker = on  
  
# Forward sound input to server:  
# microphone = disabled  
# microphone = on  
# microphone = on: Built-in Audio Analog Stereo  
# microphone = off: Built-in Audio Analog Stereo  
microphone = off  
  
# Sound Encoding  
  
# Sound plugin to use for capturing sound:  
#sound-source = pulse  
#sound-source = alsa  
#sound-source = auto  
  
# Codec(s) to use for forwarding speaker sound:  
#speaker-codec = mp3  
#speaker-codec = flac  
#speaker-codec = wav  
#speaker-codec = wavpack  
#speaker-codec = speex  
#speaker-codec = opus  
  
# Codec(s) to use for forwarding microphone sound:  
#microphone-codec = mp3  
#microphone-codec = flac  
#microphone-codec = wav  
#microphone-codec = wavpack  
#microphone-codec = speex  
#microphone-codec = opus  
  
# Try to synchronize audio and video:  
#av-sync = off  
av-sync = on
```

C.9 /etc/xpra/conf.d/30_picture.conf

 /etc/xpra/conf.d/30_picture.conf

```
#####
# Picture Encoding

# Encodings allowed:
# (not all encodings may be available in your environment):
#encodings = h264, vp8, png, png/P, png/L, webp, rgb, jpeg, h265, vp9
#encodings = all
#encodings = rgb
encodings = all

# Default encoding
# (not all encodings may be available in your environment):
#encoding = h264
#encoding = vp8
#encoding = png
#encoding = jpeg
#encoding = rgb
#encoding = webp
encoding = auto

# Used by the server to encode video:
# video-encoders = x264, vpx, nvenc
# video-encoders = none
#video-encoders = all
video-encoders = x264
#video-encoders = nvenc

# Used by both the client and server for colourspace conversion:
# csc-modules = swscale, cython, libyuv
# csc-modules = none
# csc-modules = all
csc-modules = all

# Used by the client for decoding:
# video-decoders = avcodec2, vpx
# video-decoders = none
# video-decoders = all
video-decoders = all

# Automatic video downscaling:
# video-scaling = 0           #same as off
# video-scaling = off
# video-scaling = auto       #use quality and speed settings
# video-scaling = on         #same as auto
# video-scaling = 10        #mild automatic downscaling
# video-scaling = 100       #very aggressive downscaling
```

```
video-scaling = auto

# Use fixed quality
# (value is a percentage or "auto"):
#quality = 80
quality = 100
#quality = auto

# For auto quality only:
min-quality = 30

# Use fixed speed
# (value is a percentage or "auto"):
#speed = 90
speed = auto

# For auto speed only:
min-speed = 30

# Idle delay in seconds before doing an automatic lossless refresh:
auto-refresh-delay = 0.15

# Use a fixed DPI:
#dpi = 96
#automatic (which is the default):
#dpi = 0

# Bit depth of the virtual display or client display:
# (this will be added to the xvfb command line above as "-depth VALUE
  ")
# for automatic mode, use the value 0:
#pixel-depth = 0
#pixel-depth = 16
#pixel-depth = 24
#pixel-depth = 30
pixel-depth = 0

# Video encoders loaded by the server
# (all of them unless specified)
# examples:
#video-encoders=x264,vpx,nvenc
#video-encoders=x264

# Colourspace conversion modules loaded by the server
# (all of them unless specified)
# examples:
#csc-modules=swscale,cython,opencl
#csc-modules=swscale
```

C.10 /etc/xpra/conf.d/35_webcam.conf

 /etc/xpra/conf.d/35_webcam.conf

```
#####  
# Webcam forwarding  
  
# webcam = auto  
# webcam = no  
# webcam = /dev/video0  
webcam = no
```

C.11 /etc/xpra/conf.d/40_client.conf

 /etc/xpra/conf.d/40_client.conf

```
#####  
# Client Options  
  
# OpenGL accelerated rendering:  
#opengl = yes  
#opengl = no  
#opengl = gtk, native  
#opengl = auto  
opengl = probe  
  
# How we handle server authentication requests:  
#challenge-handlers=none  
#challenge-handlers=env,file  
#challenge-handlers=env,file,kerberos,gss,u2f,prompt  
challenge-handlers=all  
  
# Client window title:  
title = @title@ on @hostinfo@  
  
# Use a header bar with controls:  
#headerbar = no  
headerbar = yes  
  
# Icon used by the system tray:  
#tray-icon = /path/to/icon.png  
  
# Client ssh command:  
#ssh = "C:\Program Files\Plink.exe" -ssh -noagent  
#ssh = /usr/bin/ssh  
#ssh = ssh -o ControlMaster=no -o ConnectTimeout=20  
#ssh = ssh -x  
#ssh = paramiko  
ssh = auto  
  
# Mouse wheel:  
#mousewheel=invert  
#mousewheel=invert-x  
#mousewheel=invert-y  
#mousewheel=off  
#mousewheel=on  
mousewheel=on  
  
# Border:  
#border=auto:off,5  
#border=auto,5  
#border=red,1
```

```
#border=no

# What to do when the user closes the window:
# - forward the action to the server:
# window-close = forward
# window-close = ignore
# window-close = disconnect
window-close = auto

# Desktop Scaling:
# - double the size of all windows:
#desktop-scaling = 2
# - increase the size by 50%:
#desktop-scaling = 1.5
#desktop-scaling = 3/2
# - disable all desktop scaling:
#desktop-scaling = off
#desktop-scaling = 0
# - enable desktop scaling and automatically select
#   a scaling value if the client display is large:
#desktop-scaling = auto
# - enable desktop scaling, but start with no scaling activated:
#desktop-scaling = 1
#desktop-scaling = on
desktop-scaling = auto

# Make desktop or shadow servers show up in fullscreen mode:
#desktop-fullscreen = yes
desktop-fullscreen = no
```

C.12 /etc/xpra/conf.d/42_client_keyboard.conf

 /etc/xpra/conf.d/42_client_keyboard.conf

```
#####
# Client Keyboard Options

# Keyboard synchronization:
keyboard-sync = yes

# Raw mode
# (unsafe: can only be used when both the client and server use the
#   same keycodes)
# keyboard-raw = no

# The options below use X11 names and semantics, on all platforms

# Layout
# (will be detected automatically, this can be used to override)
# keyboard-layout = us
# keyboard-layout = us,gb,fr

# Layouts: the different types of keyboard layouts that may be used
# (will be detected automatically, this can be used to override)
# keyboard-layouts = us,fr
# keyboard-layouts = us,gb,fr

# Variant:
# variant = ,dvorak
# variant = altgr-intl

# Default set of modifiers required by the key shortcuts:
# (can be referred to as '#' in key-shortcuts)
# shortcut-modifiers = Meta+Shift
# shortcut-modifiers = Control+Shift
# shortcut-modifiers = None
shortcut-modifiers = auto

# Key Shortcuts:
key-shortcut = Control+Menu:toggle_keyboard_grab
key-shortcut = Shift+Menu:toggle_pointer_grab
key-shortcut = Shift+F11:toggle_fullscreen
key-shortcut = #+F1:show_menu
key-shortcut = #+F2:show_start_new_command
key-shortcut = #+F3:show_bug_report
key-shortcut = #+F4:quit
key-shortcut = #+F5:show_window_info
```

```
key-shortcut = #+F10:magic_key
key-shortcut = #+F11:show_session_info
key-shortcut = #+F12:toggle_debug
key-shortcut = #+plus:scaleup
key-shortcut = #+minus:scaledown
key-shortcut = #+underscore:scaledown
key-shortcut = #+KP_Add:scaleup
key-shortcut = #+KP_Subtract:scaledown
key-shortcut = #+KP_Multiply:scalereset
key-shortcut = #+bar:scalereset
key-shortcut = #+question:scalingoff
```


C.13 /etc/xpra/conf.d/50_server_network.conf

 /etc/xpra/conf.d/50_server_network.conf

```
#####
# Server Network Options:

# broadcast servers using mDNS:
#mdns = yes
mdns = no

#####
# local unix domain sockets:

# Where to create the sockets:
# (can be specified multiple times to create multiple sockets,
#  either a directory or a socket filename)
#bind=none
#bind=auto
#bind=~/.xpra/
#bind=FILENAME
#bind=/path/to/socketfilename
#bind=/run/user/$UID/xpra/
bind = auto

# Authentication module to use for local sockets:
#auth=fail
#auth=reject
#auth=allow
#auth=file
#auth=multifile
#auth=sys
#auth=none

#####
# port number constraints:
# when starting as root for another user via the uid and gid options,
# which port numbers are allowed
min-port = 1024
#min-port = 10

#####
# TCP:

# To listen on TCP sockets:
#bind-tcp=:10000
#bind-tcp=0.0.0.0:10000
#bind-tcp=192.168.0.1:10000
```

```
# Authentication module to use for TCP sockets (see 'auth'):  
#tcp-auth=none  
#tcp-auth=sys  
#tcp-auth=sqlite:filename=/tmp/xpra-auth.sdb  
  
#####  
# SSL:  
# (see also 12_ssl.conf)  
  
# To listen on an SSL socket:  
#bind-ssl=:10001  
#bind-ssl=:443  
#bind-ssl=192.168.0.1:10001  
  
# To secure SSL sockets (see 'auth'):  
#ssl-auth=none  
  
# To support SSL on TCP sockets:  
# (requires a certificate)  
#ssl = yes  
#ssl = yes  
ssl = auto  
  
#####  
# VSOCK:  
  
# To listen on AF_VSOCK sockets:  
#bind-vsock=auto:2000  
#bind-vsock=2:2000  
  
# Authentication to use for VSOCK:  
#vsock-auth=none  
  
#####  
# html / tcp-proxy:  
  
# Where to send non xpra clients:  
# (can be used to share the port with a web server)  
#tcp-proxy = 127.0.0.1:80  
  
# Respond to HTTP requests on the TCP ports (requires bind-tcp):  
#html = no  
#html = yes  
#html = /var/www/htdocs/  
#html = auto  
html = no
```

```
#####  
# RFB upgrades  
# Allows RFB clients (ie: VNC) to connect to a plain TCP socket  
rfb-upgrade = 5
```

C.14 /etc/xpra/conf.d/55_server_x11.conf

 /etc/xpra/conf.d/55_server_x11.conf

```
#####
# X11 Server Environment:

# Set the _NET_WM_NAME,
# used by some application that make too many assumptions (ie: Java)
# To workaround JDK6 window positioning issues, use:
#wm-name = Sawfish
wm-name = Xpra

# Input methods
# To disable input method completely:
#input-method=none
# To keep the environment unchanged:
#input-method=keep
# Other possible options:
#input-method=IBus
#input-method=SCIM
#input-method=uim
input-method=none

# Paint the windows on the xvfb so that
# screen grabs and screen recorders will work
# (repaint delay in milliseconds, 0 to disable)
# sync-xvfb = 200
# sync-xvfb = 50
sync-xvfb = 0

# Fake Xinerama:
#fake-xinerama=no
#fake-xinerama=auto
#fake-xinerama=/path/to/libfakeXinerama.so
fake-xinerama = no

# Virtual display command:
# - Xvfb option (limited DPI support)
# xvfb = Xvfb -nolisten tcp -noreset \
#       +extension GLX +extension Composite \
#       -auth $XAUTHORITY \
#       -screen 0 8192x4096x24+32
# - Xephyr (requires a running X11 server):
# xvfb = Xephyr -nolisten tcp -noreset \
#       +extension GLX +extension Composite \
#       -auth $XAUTHORITY \
#       -screen 8192x4096x24+32
# - Xdummy (better with DPI patch):
# xvfb = /usr/bin/Xorg -noreset -nolisten tcp \
```

```
#      +extension GLX +extension RANDR +extension RENDER \  
#      -auth $XAUTHORITY \  
#      -logfile auto/Xorg.${DISPLAY}.log \  
#      -configdir ${XDG_RUNTIME_DIR}/.xpra/xorg.conf.d/$PID \  
#      -config /etc/xpra/xorg.conf  
#  
# Selecting virtual X server:  
xvfb = Xvfb +extension GLX +extension Composite -screen 0 \  
      8192x4096x24+32 -nolisten tcp -noreset -auth $XAUTHORITY \  
      -dpi 96
```

C.15 /etc/xpra/conf.d/60_server.conf

 /etc/xpra/conf.d/60_server.conf

```
#####
# Server Options:

# Directory for server log files:
log-dir = /run/user/$UID
#log-dir = ~/.xpra
#log-dir = /tmp

# Log file:
log-file = $DISPLAY.log

# Delegate "start" and "start-desktop" subcommands to the system-wide
  proxy server instance
start-via-proxy = auto
#start-via-proxy = no

# Scripts that can modify the environment of the server:
# source = /etc/xpra/server.env

# Scripts that can modify the environment of the commands started by
  the server:
# source-start = /etc/profile

# Environment variables for children started
# with 'start-child' or 'start':
# (can be specified multiple times)
#start-env = HOME=/home/alt
#start-env = GDK_DEBUG=dnd
start-env = #avoid Ubuntu's global menu, which is a mess and cannot be
  forwarded:
start-env = UBUNTU_MENUPROXY=
start-env = QT_X11_NO_NATIVE_MENUBAR=1
start-env = #fix for MainSoft's MainWin buggy window management:
start-env = MWNOCAPTURE=true
start-env = MWNO_RIT=true
start-env = MWM=allwm
start-env = #force GTK3 applications to use X11 so we can intercept
  them:
start-env = GDK_BACKEND=x11
start-env = #force Qt applications to use X11 so we can intercept them
  :
start-env = QT_QPA_PLATFORM=xcb
start-env = #disable Qt scaling:QT_AUTO_SCREEN_SET_FACTOR=0
start-env = QT_SCALE_FACTOR=1
start-env = #overlay scrollbars complicate things:
  GTK_OVERLAY_SCROLLING=0
```

```
start-env = #some versions of GTK3 honour this option, sadly not all:
start-env = GTK_CSD=0
start-env = #silence some AT-SPI and atk-bridge warnings:
start-env = NO_AT_BRIDGE=1

env = LD_PRELOAD=/lib/x86_64-linux-gnu/libm.so.6

# Commands to start by default
# (may be specified more than once):
# examples:
#start-child = /usr/bin/xterm
#start-child = skydel-sdx
#start-child = python3 -c 'import time; time.sleep(5)' && /usr/bin/
    firefox
#start=/bin/true
#
# Xsession can take care of initializing dbus, keyring-daemon,
# gpg-agent or whatever else might be usually started together with X
#start = /etc/X11/Xsession true
# Or you can specify individual actions:
#xrdp -merge /etc/X11/Xresources
#xrdp -merge ~/.Xresources
#
# The difference between start and start-child is only relevant
# when using exit-with-children.
# This causes the server to terminate when the last child has exited:
exit-with-children = no

# The server will terminate when the last client disconnects.
exit-with-client=yes
#exit-with-client=no

# Commands to start immediately after the first client has connected:
# (see above for the start vs start-child distinction)
#start-after-connect = touch ~/.xpra/connect
#start-child-after-connect = xterm

# Commands to every time a client connects:
# (see above for the start vs start-child distinction)
#start-on-connect = touch ~/.xpra/login
#start-child-on-connect = xterm

# Use one start-child for each user. If multiple, it also executes the
    command multiple times
start-child-on-connect = bash -c 'xhost + && if [ $(whoami) = "staff"
    ]; then xterm; fi & sudo -u simulator -g $(whoami) -i skydel-sdx'
#max-size = 1450x931

# Commands to run when the last client exits:
#start-on-last-client-exit = xterm
```

```
#start-child-on-last-client-exit = xterm

# Start a dbus-daemon:
# dbus-launch =
# dbus-launch = /usr/bin/dbus-launch --close-stderr
dbus-launch = dbus-launch --sh-syntax --close-stderr

# Start a dbus server which can be used to interact with the server
  process:
#dbus-control = no
dbus-control = yes

# Forward client dbus rpc requests:
# (requires client configuration)
#dbus-proxy = no
dbus-proxy = yes

# A wrapper for executing all sub-commands:
# exec-wrapper = vglrun --
# exec-wrapper = vglrun -d :1 --

# Allows clients to start new commands in the server context:
#start-new-commands = no
start-new-commands = no

# Start a pulseaudio server with each session:
pulseaudio = auto

# pulseaudio server start command:
pulseaudio-command = pulseaudio --start -n --daemonize=false --system=
  false --exit-idle-time=-1 --load=module-suspend-on-idle '--load=
  module-null-sink sink_name="Xpra-Speaker" sink_properties=device.
  description="Xpra\ Speaker"' '--load=module-null-sink sink_name="
  Xpra-Microphone" sink_properties=device.description="Xpra\
  Microphone"' '--load=module-native-protocol-unix socket=
  $XPRA_PULSE_SERVER' --load=module-dbus-protocol --load=module-x11-
  publish --log-level=2 --log-target=stderr --enable-memfd=no

# commands used to configure the pulseaudio server:
# pactl set-source-volume SomeSource 20%
pulseaudio-configure-commands = pactl set-default-sink Xpra-Speaker
pulseaudio-configure-commands = pactl set-default-source Xpra-
  Microphone.monitor

# Systemd specific options:
# Wrap server start commands with systemd-run:
#systemd-run = no
systemd-run = auto
```



```
#systemd-run = yes

# Command line arguments passed to systemd-run:
#systemd-run-args = -p MemoryAccounting=true -p MemoryHigh=384M -p
    MemoryLimit=512M
#systemd-run-args = -p CPUAccounting=true -p CPUQuota=25%
```

C.16 /etc/xpra/conf.d/65_proxy.conf

 /etc/xpra/conf.d/65_proxy.conf

```
#####  
# Proxy Options  
  
# Can the proxy start new sessions on demand?  
proxy-start-sessions = True  
  
# The video encoders that the proxy will claim:  
# (so the video encoding will happen in the proxy process  
# for these encoders)  
#proxy-video-encoders = nvenc  
#proxy-video-encoders = all  
proxy-video-encoders = none
```