

TIKOS

BUSINESS SOLUTIONS

# AppServer Installation information



## General note

This document describes numerous requirements and information for successfully installing the AppServer. It does not replace the minimum system requirements!

## Reverse Proxy

If a reverse proxy is to be used, the following requirements must be met:

- It is recommended to use either Apache or Nginx.
- The reverse proxy should use HTTPS for communication with the AppServer.
- It must be possible to establish HTTP connections to the AppServer and send HTTPS requests to it.
- It must be possible to establish gRPC (HTTP/2) connections to the AppServer and send gRPC requests to it.
- If Nginx is used, the settings "grpc\_set\_header Host" and "proxy\_set\_header Host" must not be used, as these can cause issues with IIS.

Configuration examples:

### Apache:

The following modules must be enabled: mod\_headers, mod\_ssl, mod\_http2, mod\_proxy, and mod\_proxy\_http2.

```
<VirtualHost *:443>
  # TODO: use correct external domain.
  ServerName appserver.localhost
  Protocols h2 http/1.1

  SSLEngine on
  SSLProxyEngine on
  # TODO: use correct certificate paths.
  SSLCertificateFile "/path/to/certificate.crt"
  SSLCertificateKeyFile "/path/to/certificate.key"

  # TODO: use correct internal ip/domain and port of appserver.
  ProxyPass / h2://192.168.240.198:443/ keepalive=On upgrade=websocket
  ProxyPassReverse / h2://192.168.240.198:443/
</VirtualHost>
```

### Nginx:

The settings grpc\_set\_header Host and proxy\_set\_header Host must not be configured, as they can cause errors with IIS (for example, 400 Bad Request errors).

When using grpc\_ssl\_verify or proxy\_ssl\_verify, it must be ensured that the corresponding settings (e.g., accepted certificates) are configured correctly.

```
upstream appserver {
    # TODO: use correct internal ip/domain and port of appserver.
    server 192.168.240.198:443;
}

server {
    listen 443 ssl;
    http2 on;
    # TODO: use correct external domain.
    server_name appserver.localhost;

    # TODO: use correct certificate paths.
    ssl_certificate    /path/to/certificate.crt;
    ssl_certificate_key /path/to/certificate.key;

    location / {
        grpc_connect_timeout 10s;
        grpc_read_timeout    300s;
        grpc_send_timeout     300s;

        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $http_connection;
        proxy_http_version 1.1;

        # Nginx does only allow http2 upstream connections for grpc requests.
        # But the AppServer does also provide for example a normal http api which would break if we
        use grpc_pass for everything.
        # Because of this we need to check the content type and decide what to do with the request.
        if ($http_content_type = "application/grpc") {
            grpc_pass grpcs://appserver;
        }

        proxy_pass https://appserver;
    }
}
```