Configuring Teamwork Cloud with a Proxy

On this page:

Scripts

- nGinx-Proxy-Pkg.zip
- HAProxy-Pkg.zip

⚠ Please note that this information is provided as a courtesy only and support services are not offered for any of the features described in this article.

Introduction

There are environments in which Teamwork Cloud applications need to be fronted by a proxy. The most widespread use case for this is port restrictions, where the native ports cannot be exposed. Typically, all external traffic is restricted to a single port, such as 443, which is allowed to traverse corporate firewalls or proxies.

To configure a proxy, you first need to understand the traffic flows, since each traffic flow needs to be addressed. Teamwork Cloud is composed of two services (Webapp and Teamwork Cloud), which need to expose three traffic flows (or port bindings) to function.

- Webapp (native port 8443 - http/s)
- Teamwork Cloud
  - REST API (native port 8111 - http/s)
  - Client Communication (native port 3579 -TCP cleartext or native port 10002 - TLS/TCP encrypted) - 10002 is the default port used by the client and is configured in the Teamwork Cloud Admin Settings page

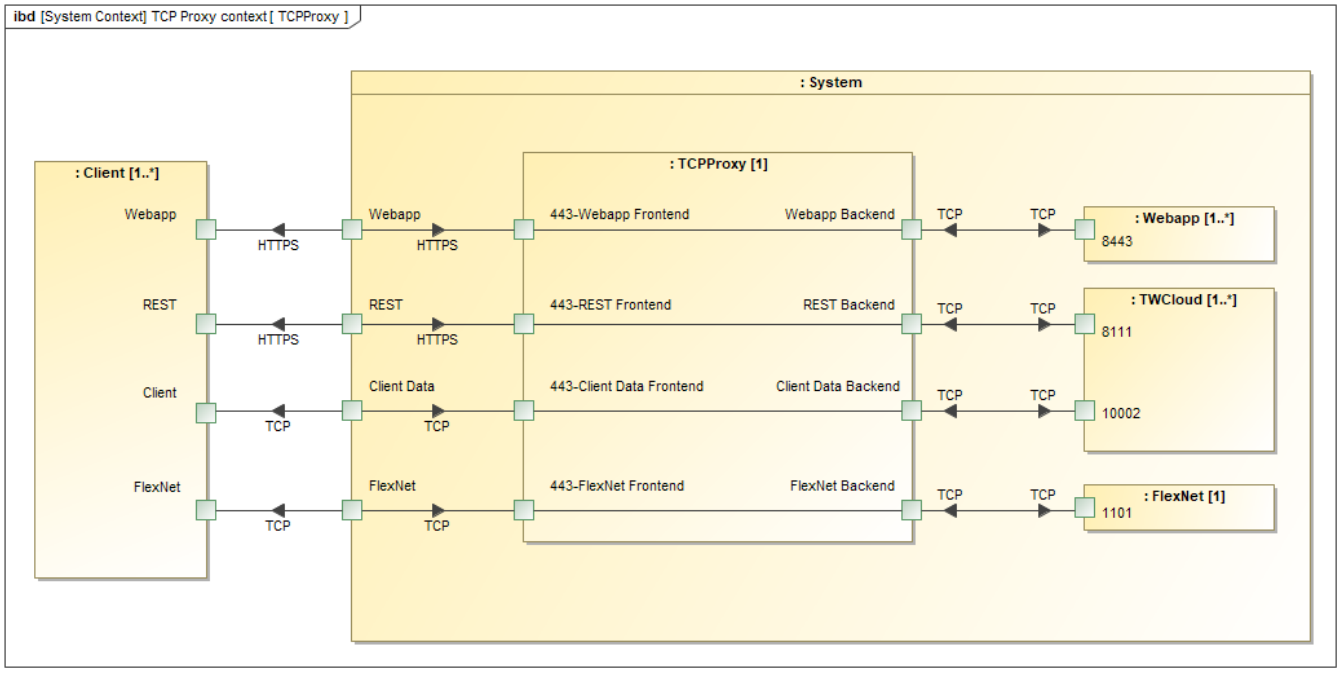
Additionally, if the FlexNet license server is running on the same instance with the same port constraints, a TCP proxy must be created for it, forwarding to our cameo vendor daemon (native port 1101 - TCP).

Because you can only bind a single instance of a port to an IP address in TCP/IP, the instance will need to have multiple IP addresses in which to bind each traffic flow. Traffic flows are tied to frontends (the part of the proxy exposed to the external world that receives requests and forwards them to the backends) and backends (handling of the actual requests). The number of IP addresses required depends on the type of proxying you configure.

There are two types of proxying: TCP (the proxying is done at Layer 4 of the OSI model); or HTTP (the proxying is done at Layer 7 of the OSI).

Layer 4 (TCP) Proxy

Layer 4 proxying is the lighter-weight of the 2 methods as it simply forwards incoming packets from the frontend to its associated backend. In this case, you need 3 IP addresses (4 if proxying FlexNet as well) in order to bind to each data flow.

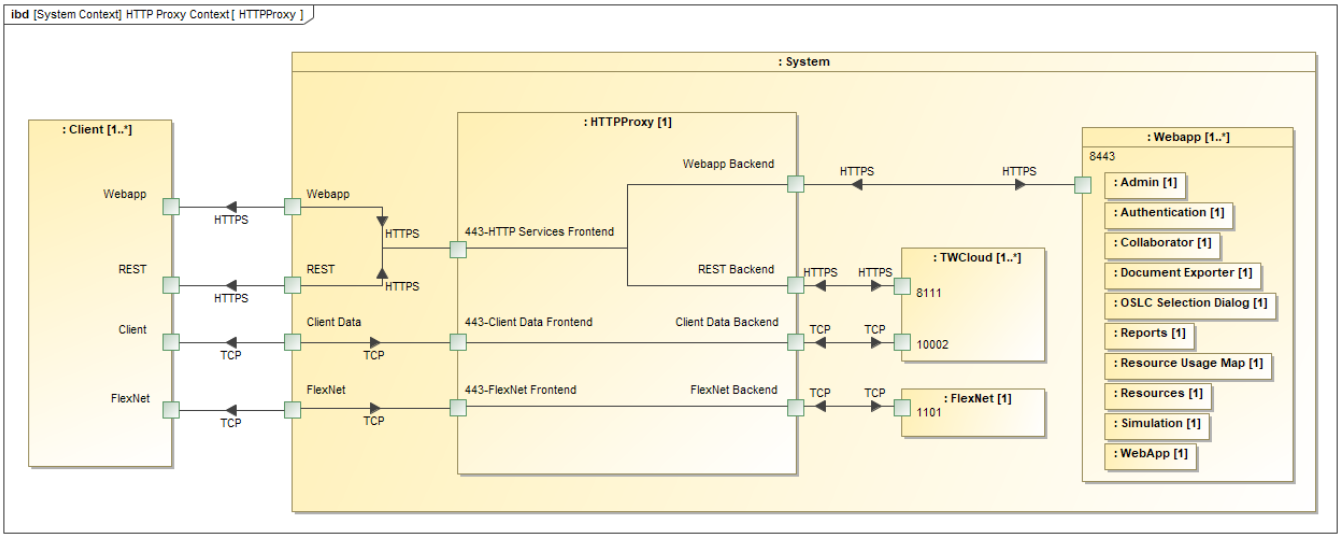


Layer 4 proxy configuration example.

Layer 7 (HTTPS) Proxy

Layer 7 proxying inspects the actual content of the data coming through the proxy. Therefore, SSL termination takes place at the proxy, so you can now manipulate the data coming through and take specific actions. Since each of the HTTPS services exposes a path (/webapp for authentication and Web Platform applications, /osmc for REST API), you can treat the incoming data as a single flow (a single frontend), and have the proxy send the request to the respective backend based on the path of the request.

Please note that the diagram below depicts a hybrid layer 7 configuration.



Layer 7 proxy configuration example.

There are 2 classes of proxies/load balancers: hardware (such as F5 Big-IP, Cirtix NetScaler) which is external to the application instance; and software (such as nGinx, HAProxy) which can be run externally (on a dedicated instance) or on the same instance as the application.

Proxy configuration

This section provides an outline and examples of how to deploy both Layer 4 (TCP) and Layer 7 Hybrid (HTTPS) proxies for nGinx and HAProxy. The Layer 7 configuration is a hybrid configuration in that it also includes Layer 4 proxying for the modeling tool client data stream and FlexNet server. Prepackaged configuration scripts and template files are provided here to generate the basic configuration files.

Teamwork Cloud is initially deployed with a local IP address. This will become the IP address for the Webapp frontend. Keep in mind that the native ports (8443, 8111, clientTeamwork Cloud port, and the cameo vendor daemon for FlexNet) bind to all interfaces.

The example configuration packages use four IP addresses - 10.254.254.31 (Webapp), 10.254.254.32 (REST), 10.254.254.34 (clientTeamwork Cloud), and 10.254.254.35 (FlexNet). These are the most basic configurations that allow the system to operate. You may add features or change behaviors by modifying these files.

Note that to simplify all of the aspects of certificates, you should either use a wildcard certificate or one that contains SANs for all of the public FQDNs and /or IP addresses.

In a Layer 7 Hybrid HTTPS proxy, SSL termination occurs at the proxy. A PEM-encoded file containing the private key and certificates is needed for the server, including the full certificate chain.

Teamwork Cloud Configuration

Update the following Teamwork Cloud configuration files after deploying your proxy. This applies to both Layer 4 TCP and Layer 7 HTTPS.

Append to an *authserver.properties* whitelist to allow the proxied Teamwork Cloud web access.

authserver.properties

```
# authentication.redirect.uri.whitelist needs to have the Webapp's URL appended to the string.  If using port
443 as the public port for WebApp, the port number is omitted
authentication.redirect.uri.whitelist=https://10.254.254.31:8443/,https://10.254.254.31:8111/,
https://md_redirect,https://10.254.254.31/
```

Update authentication server address and port in *webappplatform.properties*.

webappplatform.properties

```
# Specify the WebApp authentication server location.
# IP address or domain name.
authentication.server.ip=10.254.254.31
# Specify the Authentication server port.
authentication.server.port=443
```