

## NightFall API Client Documentation

The NightFall API client is a HTTP client implemented using [curl](#) and [curlpp](#).

It allows modders to communicate to the outside world via HTTP/JSON.

### Cvar Settings

#### *sv\_api\_client*

Name	sv_api_client
Allowed values	0 or 1
Description	Enables or disable API client. This setting is only affected at map start. It's value is effective at map start/change/restart. API client can't be started or stopped in the middle of the map. If you change it's value, change the map or restart the map.

### Client Behaviour

#### Initialization

Once [sv\\_api\\_client](#) cvar is set to one, and upon map load, API client is started. API client will have a single idle running worker thread to serve the HTTP client requests.

#### Run-time

Whilst the API client is running, modder can choose to register a callback script that will handle outgoing requests. See [create\\_api\\_request](#) and [Script Usage](#) for more info.

Once the request is done, callback function is called.

Please note that this process is not fast, since everything is synced with MOHAA's 20 fps server.

Usual calls made when a request arrives are as follows:

1. Modder calls [create\\_api\\_request](#) via script.
2. MOHAA thread **waits/syncs** to take the send request/script and pass it to API client.
3. API client thread **waits/syncs** to get request from MOHAA thread.
4. API client thread creates HTTP request.
5. Once HTTP request is finally done, API client thread **waits/syncs** to send response to MOHAA thread.
6. MOHAA thread **waits/syncs** to take the next response passed by API client. Allowed rate is one response per frame.
7. MOHAA thread calls script callback for the HTTP request.

Since this process is heavy for MOHAA thread, it was decided to make it at a minimal load. Therefore, limitation of one response per frame at step 6 is made.

### Shutdown

At map end(or map change or map restart), API client will shutdown, clear all pending requests/responses and scripts.

---

### Script Usage

When a user creates an API request using [create\\_api\\_request](#), the script callback function should be something like this:

```
request_handler local.http_code local.resp local.user_data:
    println "~~~~~"
    println "http_code: " local.http_code
    println "resp: " local.resp
    println "user_data: " local.user_data
    println "~~~~~"
end
```

`local.http_code` is the HTTP response code.

`local.resp` is the string containing HTTP response body.

`local.user_data` is the user-specific custom data specified when calling [create\\_api\\_request](#).

If an error occurs during the request, `local.http_code` is set to -1 and error string is saved inside `local.resp`.