

# Administrator's Guide

WebtoB 6.1

**TMAXSOFT**

# Copyright

Copyright 2025. TmaxSoft Co., Ltd. All Rights Reserved.

## Company Information

TmaxSoft Co., Ltd.

TmaxSoft Tower, Jeongjail-ro 45, Bundang-gu, Seongnam-si, Gyeonggi-do 13613, South Korea

Website: <https://www.tmaxsoft.com/en/>

## Restricted Rights Legend

All TmaxSoft Software (Tmax WebtoB®) and documents are protected by copyright laws and international convention. TmaxSoft software and documents are made available under the terms of the TmaxSoft License Agreement and this document may only be distributed or copied in accordance with the terms of this agreement. No part of this document may be transmitted, copied, deployed, or reproduced in any form or by any means, electronic, mechanical, or optical, without the prior written consent of TmaxSoft Co., Ltd. Nothing in this software document and agreement constitutes a transfer of intellectual property rights regardless of whether or not such rights are registered) or any rights to TmaxSoft trademarks, logos, or any other brand features.

This document is for information purposes only. The company assumes no direct or indirect responsibilities for the contents of this document, and does not guarantee that the information contained in this document satisfies certain legal or commercial conditions. The information contained in this document is subject to change without prior notice due to product upgrades or updates. The company assumes no liability for any errors in this document.

## Trademarks

WebtoB® is a registered trademark of TmaxSoft Co., Ltd. Other products, titles or services may be registered trademarks of their respective companies.

## Open Source Software Notice

Some modules or files of this product are subject to the terms of the following licenses: zlib License, Apache License 2.0, Boost Software License 1.0, BSD 3-Clause License, MIT License, curl License (MIT-like), LGPL-2.1-or-Later, BSD License (BSD-3-Clause)

Detailed Information related to the license can be found in the following directory:

`${INSTALL_PATH}/lib/licenses`

## Maintenance

| Category                                | Support Item                 | Service Description                                                                                                                                                                                                                                                                                                                  |
|-----------------------------------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Product Support                         | Patches & Upgrades           | Complimentary patch service<br><br>Discount for major version upgrades<br><br>Patch history via web support                                                                                                                                                                                                                          |
| Technical Support<br>- Basic Service    | Incident Support             | Cause analysis and resolution upon incidents<br><br>Three-tier incident support: Service Desk Team → Technical Team → R&D                                                                                                                                                                                                            |
|                                         | Routine Support (Online)     | Q&A services through online resources such as E-mail, phone, remote access, and website                                                                                                                                                                                                                                              |
|                                         | Customized Support (On-site) | On-site support service provided upon customer request                                                                                                                                                                                                                                                                               |
| Technical Support<br>- Optional Service | Preventive Support           | System operation reporting and incident prevention through regular inspections<br><br><ul style="list-style-type: none"><li>◦ Gathers requirements from administrators or operations.</li><li>◦ Provides operation reports (system and engine operations).</li><li>◦ Reports system improvement recommendations as needed.</li></ul> |
| Maintenance Cost & Duration             | TBD upon contract            | EOL/EOS documents provided at contract signing                                                                                                                                                                                                                                                                                       |

## Document History

| Product Version | Guide Version | Date       | Remarks               |
|-----------------|---------------|------------|-----------------------|
| WebtoB 6.1      | 3.1.2         | 2025-12-03 | Added WJP limitations |
| WebtoB 6.1      | 3.1.1         | 2025-11-21 | GA Release            |

# Contents

|                                                      |    |
|------------------------------------------------------|----|
| 1. Getting Started .....                             | 1  |
| 1.1. License Specific Functions .....                | 1  |
| 1.2. Quick Start .....                               | 1  |
| 2. Startup and Shutdown .....                        | 6  |
| 2.1. Starting WebtoB .....                           | 7  |
| 2.2. Stopping WebtoB .....                           | 7  |
| 3. Configuration .....                               | 8  |
| 3.1. Basic Structure .....                           | 8  |
| 3.1.1. Types of Configurable Sections .....          | 10 |
| 3.1.2. Setting Value Formats and Configuration ..... | 11 |
| 3.2. ACCESS Section .....                            | 12 |
| 3.2.1. Configuration Items .....                     | 12 |
| 3.2.2. Default Settings .....                        | 15 |
| 3.2.3. Example .....                                 | 15 |
| 3.3. ALIAS Section .....                             | 15 |
| 3.3.1. Configuration Items .....                     | 16 |
| 3.3.2. Example .....                                 | 17 |
| 3.4. DESTINATION Section .....                       | 17 |
| 3.4.1. Configuration Items .....                     | 17 |
| 3.4.2. Default Settings .....                        | 56 |
| 3.4.3. Example .....                                 | 56 |
| 3.5. DIRECTORY Section .....                         | 57 |
| 3.5.1. Configuration Items .....                     | 57 |
| 3.5.2. Example .....                                 | 60 |
| 3.6. ERRORDOCUMENT Section .....                     | 61 |
| 3.6.1. Configuration Items .....                     | 61 |
| 3.6.2. Example .....                                 | 62 |
| 3.7. FILTER Section .....                            | 63 |
| 3.7.1. Configuration Items .....                     | 63 |
| 3.7.2. Example .....                                 | 65 |
| 3.8. HEADERS Section .....                           | 71 |
| 3.8.1. Configuration Items .....                     | 71 |
| 3.8.2. Example .....                                 | 74 |
| 3.9. LOGGING Section .....                           | 74 |
| 3.9.1. Configuration Items .....                     | 74 |
| 3.9.2. Example .....                                 | 92 |
| 3.10. NODE Section .....                             | 93 |
| 3.10.1. Configuration Items .....                    | 93 |

|                                                                |     |
|----------------------------------------------------------------|-----|
| 3.10.2. Example .....                                          | 108 |
| 3.11. SERVER Section .....                                     | 108 |
| 3.11.1. Configuration Items .....                              | 108 |
| 3.11.2. Example .....                                          | 144 |
| 3.12. SERVICE section .....                                    | 145 |
| 3.12.1. Configuration Items .....                              | 145 |
| 3.12.2. Default Behavior .....                                 | 154 |
| 3.12.3. Example .....                                          | 154 |
| 3.13. SSL Section .....                                        | 155 |
| 3.13.1. Configuration Items .....                              | 155 |
| 3.13.2. Example .....                                          | 164 |
| 4. WebtoB Process Structure .....                              | 166 |
| 4.1. Process Types .....                                       | 166 |
| 4.2. Manager Process .....                                     | 166 |
| 4.2.1. Retrieving Configuration (/config) .....                | 166 |
| 4.2.2. Shutting Down WebtoB Manager (/shutdownmanager) .....   | 167 |
| 4.3. Engine Process .....                                      | 168 |
| 4.4. Process Privileges and Privileged Ports .....             | 169 |
| 4.4.1. Privileged Ports and WebtoB Execution Permissions ..... | 169 |
| 4.4.2. Issues When Running as Root .....                       | 169 |
| 4.4.3. Engine Privilege Separation .....                       | 169 |
| 5. URL Rewrite .....                                           | 171 |
| 5.1. Configuting URLRewrite Feature .....                      | 171 |
| 5.2. Defining Rewriting Conditions .....                       | 171 |
| 5.2.1. TestString Configuration .....                          | 171 |
| 5.2.2. CondPattern Configuration .....                         | 172 |
| 5.2.3. Flags Configuration .....                               | 173 |
| 5.3. Defining Rewriting Behavior .....                         | 174 |
| 5.3.1. Pattern Configuration .....                             | 174 |
| 5.3.2. Substitution Configuration .....                        | 174 |
| 5.3.3. Flags Configuration .....                               | 175 |
| 5.4. Examples of URL Rewrite Configuration .....               | 177 |
| 6. WebtoB Admin API .....                                      | 181 |
| 6.1. Environment Information .....                             | 183 |
| 6.1.1. /config .....                                           | 183 |
| 6.1.2. /updateconfig .....                                     | 184 |
| 6.1.3. /webtobinfo .....                                       | 184 |
| 6.1.4. /patchinfo .....                                        | 185 |
| 6.2. Operational Status Information .....                      | 186 |
| 6.2.1. /cliinfo .....                                          | 186 |
| 6.2.2. /connectioninfo .....                                   | 187 |

|                                                  |     |
|--------------------------------------------------|-----|
| 6.2.3. /svginfo .....                            | 190 |
| 6.2.4. /statinfo .....                           | 191 |
| 6.2.5. /restat .....                             | 194 |
| 6.3. Process Shutdown .....                      | 195 |
| 6.3.1. /shutdown .....                           | 195 |
| 6.4. Cache Information Management .....          | 196 |
| 6.4.1. /cachelist .....                          | 196 |
| 6.4.2. /cacherefresh .....                       | 197 |
| 6.5. Certificate Management .....                | 198 |
| 6.5.1. /certrefresh .....                        | 198 |
| 7. WebtoB Console Tool .....                     | 199 |
| 7.1. wsadmin .....                               | 199 |
| 7.1.1. help .....                                | 202 |
| 7.1.2. client-info (cli, cliinfo) .....          | 204 |
| 7.1.3. connect .....                             | 205 |
| 7.1.4. patch-info (patchinfo) .....              | 205 |
| 7.1.5. webtob-info (webtobinfo, wi) .....        | 206 |
| 7.1.6. license-info (licenseinfo, li) .....      | 207 |
| 7.1.7. system-info (systeminfo) .....            | 207 |
| 7.1.8. stat-info (st, stat) .....                | 208 |
| 7.1.9. restat .....                              | 210 |
| 7.1.10. config (cfg) .....                       | 211 |
| 7.1.11. update-config .....                      | 212 |
| 7.1.12. cache-list (cachelist) .....             | 212 |
| 7.1.13. cache-refresh (cr) .....                 | 213 |
| 7.1.14. cert-refresh (cr) .....                  | 213 |
| 7.1.15. svg-info (si, sgi, svginfo) .....        | 214 |
| 7.1.16. connection-info (ci, con, coninfo) ..... | 215 |
| 7.1.17. shutdown (down) .....                    | 217 |
| 7.1.18. exit .....                               | 217 |
| 7.2. configMigrator .....                        | 217 |
| 7.3. configValidator .....                       | 219 |
| 7.4. mkpwd .....                                 | 220 |
| Appendix A: Configuration File Examples .....    | 222 |
| A.1. Basic Configuration .....                   | 222 |
| A.2. JEUS Integration Configuration .....        | 223 |
| A.2.1. WebtoB Configuration .....                | 223 |
| A.2.2. JEUS Configuration .....                  | 225 |
| A.3. Commonly Used Settings .....                | 226 |

# 1. Getting Started

## 1.1. License Specific Functions

WebtoB provides the following functions according to the license type.

| License Type | Description                                                                                                                                                                                                                                                                                                                                       |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TRIAL        | Supports only one HTH and a maximum of five users.                                                                                                                                                                                                                                                                                                |
| BASE         | Used when WebtoB is embedded in JEUS.<br><br>Supports only one HTH.<br><br>In UNIX/Linux environments, the WJP port is not used because integration with JEUS is supported only via domain sockets (named pipes).                                                                                                                                 |
| STANDARD     | One or more HTHs are supported.<br><br>Can connect to JEUS without limitations and supports most WebtoB functions.                                                                                                                                                                                                                                |
| ENTERPRISE   | Supports the following functions in addition to all of those provided in the STANDARD version. <ul style="list-style-type: none"><li>◦ Supports Reverse Proxy Group function for multi-configuration when connecting to another web application server through reverse proxy.</li><li>◦ Supports FILTERS process for filter processing.</li></ul> |

## 1.2. Quick Start

The following are simple steps to quickly verify WebtoB's normal operation of after installation:

1. Use **configValidator**, WebtoB's configuration file validation tool, to verify that the configuration file is properly set up.

```
$ configValidator
```

The following is an example of configuration of the WEBTOB\_HOME\config\webtob-config.json file.

```
{
  "node": {
    "name": "webtob_node",
    "hth_count": 1,
    "worker_threads": 8,
    "hth_schedule": "RR"
  },
  "server": {
```

```

"http": {
  "common_config": {
    "doc_root": "docs",
    "service_order": "uri,ext",
    "access_log": "access_log1"
  },
  "http_servers": [
    {
      "name": "http1",
      "port": [8080],
      "enable_ssl": true,
      "ssl_name": [
        "ssl1"
      ]
    }
  ]
},
"wjp": {
  "port": 9999,
  "wjp_servers": [
    {
      "name": "MyGroup1",
      "svr_chk_time": 60
    }
  ]
},
"logging": {
  "access_log": [
    {
      "name": "access_log1",
      "level": "INFO",
      "format": "DEFAULT",
      "handlers": {
        "file_handler": {
          "file_name": "logs/access.log"
        },
        "enable_console_handler": false
      }
    }
  ],
  "error_log": {
    "level": "INFO",
    "handlers": {
      "file_handler": {
        "file_name": "logs/errors.log"
      },
      "enable_console_handler": false
    }
  },
  "system_log": [
    {
      "name": "webtob",
      "level": "INFO",
      "handlers": {
        "file_handler": {
          "file_name": "logs/webtob.log"
        },
        "enable_console_handler": true
      }
    }
  ]
}

```

```

    }
  }
],
},
"destination": {
  "jeus": [
    {
      "name": "MyGroup1",
      "server_schedule": "RR",
      "connection_schedule": "RR"
    }
  ],
  "reverse_proxy": {
    "reverse_proxy_group": [
      {
        "name": "rproxyGroup1",
        "reverse_proxy_server": [
          {
            "address": "internal.server:80"
          }
        ]
      }
    ]
  }
}],
"htmls": [
  {
    "name": "htmls1"
  }
]
},
"service": {
  "uri": [
    {
      "name": "uri1",
      "target_http_servers": [
        "http1"
      ],
      "match": {
        "type": "prefix",
        "target": "/rproxy",
        "rewrite": "/"
      },
      "destination": {
        "type": "REVERSE_PROXY",
        "target": "rproxyGroup1"
      }
    }
  ],
  "ext": [
    {
      "name": "ext1",
      "target_http_servers": [
        "*"
      ],
      "match": {
        "type": "exact",
        "target": "text/html"
      },
      "destination": {
        "type": "JEUS",

```

```

        "target": "MyGroup1"
      }
    }
  ],
  "ssl": {
    "ssl_configs": [
      {
        "name": "ssl1",
        "webtob_certificate_file": "server.crt",
        "webtob_certificate_key_file": "key.crt"
      }
    ]
  }
}

```

The following items must be modified when creating the webtob-config.json file.

| Item                                | Description                                      |
|-------------------------------------|--------------------------------------------------|
| server > http > http_servers > port | Specifies the port used by the HTTP server.      |
| server > wjp > port                 | Specifies the port to be used by the WJP server. |



If a port number that is already in use is specified for the HTTP or WJP server, WebtoB may fail to start. Be sure to specify unique port numbers.

2. Enter the command to start WebtoB in the command prompt.

```
$ webtob
```

After starting WebtoB, you can check the names and PIDs of the running WebtoB processes using the 'ps -eo user,pid,comm | grep WebtoB' command.

The following describes the roles of each process.

| Process        | Role                                                                                                                          |
|----------------|-------------------------------------------------------------------------------------------------------------------------------|
| WebtoB Manager | Manages the WebtoB engine.<br>Automatically restarts the engine if the engine processing HTTP requests terminates abnormally. |
| WebtoB Engine  | Receives and processes HTTP requests.                                                                                         |

For a specific process, you can check the names of the running threads using the 'ps -p [WebtoB 6 PID] -L' or 'top -H -p [WebtoB 6 PID]' command.

The following describes the roles of each thread.

| Thread      | Role                                                                                                                                                                                                       |
|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BOOT        | The main thread that runs when WebtoB starts.<br>It handles the overall server booting process, including loading configurations, creating and starting the server, etc.                                   |
| API-HANDLER | The thread that handles API requests.<br>It performs WebtoB status monitoring or commands according to the client's REST API request.                                                                      |
| HTL         | The thread that accepts client connections.                                                                                                                                                                |
| HTH         | The thread that handles client requests.<br>It performs parsing of requests/responses, network I/O, and delegates specific tasks to the WORKER thread.                                                     |
| WORKER      | The thread that processes commands from the HTH thread.<br>Each WORKER is created and assigned by a specific HTH to handle tasks such as static content (HTMLS) processing. (To be expanded in the future) |
| LOGGING     | The thread responsible for handling all log messages output from WebtoB.                                                                                                                                   |

- Open a browser and enter the following URL.

```
http://<IP Address>:<8080 or a user-specified port number>/
```

If WebtoB starts without any error, the following page appears.

# WebtoB Web Server has been successfully Installed on this website!

If you can see this page, then the owner of this domain has  
successfully installed the [WebtoB Web Server](#) software.



- Enter the WebtoB shutdown command in the command prompt.

```
$ webtoB -a stop
```

## 2. Startup and Shutdown

This chapter describes the commands and various options for starting and stopping WebtoB.

To start and stop the WebtoB system, use the **webtob** binary located in the **\${WEBTOB\_HOME}/bin** directory.

- Usage

```
> webtob [-a <target action>][-c <configuration file name>][-p <configuration file path>][-t  
<forced shutdown time>][-i][-v][-V][-L][-h]
```

| Option                         | Description                                                                                                                                                                                                                        |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| [-a <target action>]           | Specifies the target action. (Default: start) <ul style="list-style-type: none"><li>◦ start</li><li>◦ start-engine</li><li>◦ start-manager</li><li>◦ stop</li><li>◦ stop-engine</li><li>◦ stop-manager</li><li>◦ restart</li></ul> |
| [-c <configuration file name>] | Specifies the configuration file name. (Default: webtob-config.json)                                                                                                                                                               |
| [-p <configuration file path>] | Specifies the configuration file path. (Default: \${WEBTOB_HOME}/config)                                                                                                                                                           |
| [-t <Forced shutdown time>]    | Specifies the timeout before a forced shutdown. (Unit: seconds, default: 60)                                                                                                                                                       |
| [-i]                           | Performs an immediate forced shutdown. By default, it completes ongoing tasks before shutdown, but using this option does not guarantee task completion.                                                                           |
| [-v]                           | Displays the current WebtoB version information.                                                                                                                                                                                   |
| [-V]                           | Starts WebtoB in verbose mode. Since logs are printed to the console and WebtoB runs as a foreground process, the shell cannot be used until the process stops.                                                                    |
| [-L]                           | Identical to <b>[-V]</b> option, but log level is fixed at DEBUG regardless of configuration.                                                                                                                                      |
| [-h]                           | Prints help information.                                                                                                                                                                                                           |

## 2.1. Starting WebtoB

To start WebtoB, run the **webtob** or **webtob -a start** command.

The following is an example of executing the **webtob** command to start WebtoB.

```
$ webtob
Environment variables for WebtoB:
    WEBTOB_HOME = /home/webtob/

Welcome to WebtoB Demo system.
Enterprise edition License in used.
Success to load config files : webtob-config.json
***** Configuration initialized successfully *****
Current WebtoB Configuration:
    Number of client handlers (HTH) = 1
    Number of worker threads per handler (HTH) = 8
    Number of connection pool size per handler (HTH) = 8192

Today is 2025-06-12
License expires on 2099-12-31 (27230 days remaining)

Booting WebtoB on node (webtob)
Current maximum open files per node 1024
Limit of open files is increased from 1024 to 1048576
Version: WebtoB 6 Fix #1 Linux B0 epoll 2025.06.05
Starting Server(pid:93736) at Thu Jun 12 16:49:14 2025
WebtoB manager starts managing WebtoB process (PID: 1446773)
```

## 2.2. Stopping WebtoB

To stop WebtoB, run the **webtob -a stop** command. Before stopping WebtoB, check the `.webtob.pid` file to see if WebtoB is currently running. Then, go to the `${WEBTOB_HOME}/bin` directory and run the **webtob -a stop** command.

The following is an example of executing the **webtob -a stop** command to stop WebtoB.

```
$ webtob -a stop
Environment variables for WebtoB:
    WEBTOB_HOME = /home/webtob/

Shutting down WebtoB
WebtoB process (PID: 93736) is terminated normally. Stop managing process.
WebtoB(pid: 93736) is stopped.
Shutting down WebtoB Manager
WebtoB Manager (PID: 1446773) is stopped.
```

# 3. Configuration

## 3.1. Basic Structure

WebtoB environment configuration defines the behavior of a single WebtoB node and the web server.

The configuration file supports **.YAML** and **.JSON** formats, as well as **.m** format used in WebtoB 5. Each setting consists of a 'section name', 'item name', and 'setting value'.

The following is the configuration format in JSON.

```
"Section Name": {  
  "Item" : Setting Value,  
  ...,  
  "Item" : Setting Value  
}
```

The start of each section is followed by the section name and then configuration items.



Configuration must be set according to the rules of JSON or YAML format, depending on the chosen format.

The following is an example of configuration in JSON format.

```
{  
  "node": {  
    "name": "webtob_node",  
    "hth_count": 1,  
    "worker_threads": 8  
  },  
  "server": {  
    "http": {  
      "common_config": {  
        "doc_root": "docs",  
        "access_log": "access_log1"  
      },  
      "http_servers": [  
        {  
          "name": "http1",  
          "port": [8080],  
          "enable_ssl": true,  
          "ssl_name": [  
            "ssl1"  
          ]  
        }  
      ]  
    }  
  }  
}
```

```

    },
    "wjp": {
      "port": 9999,
      "wjp_servers": [
        {
          "name": "MyGroup1"
        }
      ]
    }
  },
  "logging": {
    "access_log": [
      {
        "name": "access_log1",
        "level": "INFO",
        "format": "DEFAULT",
        "handlers": {
          "file_handler": {
            "file_name": "logs/access.log"
          }
        }
      }
    ],
    "error_log": {
      "level": "INFO",
      "handlers": {
        "file_handler": {
          "file_name": "logs/errors.log"
        }
      }
    },
    "system_log": [
      {
        "name": "webtob",
        "level": "INFO",
        "handlers": {
          "file_handler": {
            "file_name": "logs/webtob.log"
          }
        }
      }
    ]
  },
  "destination": {
    "jeus": [
      {
        "name": "MyGroup1"
      }
    ],
    "reverse_proxy": {
      "reverse_proxy_group": [
        {
          "name": "rproxyg1",
          "reverse_proxy_server": [
            {
              "address": "internal.server:80",
              "name": "rproxy1"
            }
          ]
        }
      ]
    }
  }
}

```

```

    }},
    "htmls": [
      {
        "name": "htmls1"
      }
    ]
  },
  "service": {
    "uri": [
      {
        "name": "uri1",
        "target_http_servers": [
          "http1"
        ],
        "match": {
          "type": "prefix",
          "target": "/rproxy",
          "rewrite": "/"
        },
        "destination": {
          "type": "REVERSE_PROXY",
          "target": "rproxyGroup1"
        }
      }
    ],
    "ext": [
      {
        "name": "ext1",
        "target_http_servers": [
          "*"
        ],
        "match": {
          "type": "exact",
          "target": "text/html"
        },
        "destination": {
          "type": "JEUS",
          "target": "MyGroup1"
        }
      }
    ]
  },
  "ssl": {
    "ssl_configs": [
      {
        "name": "ssl1",
        "certificate_file": "server.crt",
        "certificate_key_file": "key.crt"
      }
    ]
  }
}

```

### 3.1.1. Types of Configurable Sections

The following describes each section that can be configured in WebtoB.

| Section                       | Description                                                                              |
|-------------------------------|------------------------------------------------------------------------------------------|
| <a href="#">ACCESS</a>        | Client access control based on IP address, network/netmask, and header information.      |
| <a href="#">ALIAS</a>         | Alias for the physical server path and URI.                                              |
| <a href="#">DESTINATION</a>   | The internal server that will actually handle client requests.                           |
| <a href="#">ERRORDOCUMENT</a> | Error response actions.                                                                  |
| <a href="#">FILTER</a>        | Section for using the Filter module.                                                     |
| <a href="#">HEADERS</a>       | Settings for controlling HTTP request and response headers.                              |
| <a href="#">LOGGING</a>       | Logging format for client requests.                                                      |
| <a href="#">NODE</a>          | Node settings.                                                                           |
| <a href="#">SERVER</a>        | Settings for the information that clients can access.                                    |
| <a href="#">SERVICE</a>       | Settings for internal server to handle HTTP requests based on the request's URI and EXT. |
| <a href="#">SSL</a>           | SSL to use on WebtoB.                                                                    |



The [DESTINATION](#), [SERVER](#), and [SERVICE](#) sections are required.

### 3.1.2. Setting Value Formats and Configuration

The values of each item are set in one of the following types: object, integer, string, boolean, or array. The following describes each value type.

| Type    | Description                                                                                                                                                                                  |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| object  | Used when there are multiple child settings. <ul style="list-style-type: none"> <li>◦ JSON: Enclosed with curly braces ({ })</li> <li>◦ YAML: Indent one level after a line break</li> </ul> |
| integer | Numeric values.                                                                                                                                                                              |
| string  | String values.                                                                                                                                                                               |
| boolean | Sets to either true or false.                                                                                                                                                                |
| array   | Sets multiple settings of the same type. Enclose with square brackets ([ ]) and separate each setting with a comma (,), or use hyphen (-) after a line break.                                |



To fully understand the configuration items described in this guide, you must first be familiar with the basic rules as follows:

- An item starting with the hash symbol (#) before its name is an optional item.
- If a format is enclosed in square brackets ([ ]), it indicates an array format

composed of specific types.

- If a default value exists, specify the value after the hash symbol.
- For an integer variable, use parentheses to display the valid range.
- If the maximum limit does not exist, omit the range. The maximum value must be less than or equal to INT\_MAX (2147483647).
- "\$ENV" means the environment variable can be referenced.
- If "R.PATH" is specified as a relative path, it is relative to \$WEBTOB\_HOME.
- For object format, internal settings are expressed in curly brackets ( { } ).
- For object format, if the same format exists in a parent setting, the description is omitted with "{...}".
- "COMMON" indicates that if not specified, it follows the parent setting.

## 3.2. ACCESS Section

The ACCESS section configures client accesses based on IP address, network/netmask, header information, and the order of applying configurations.

This section can be applied to the SERVICE and SERVER sections. The defined resources from each of these sections are accepted or denied.

### 3.2.1. Configuration Items

The following is the configuration format of the ACCESS section.

```
#"access": {  
  "access_list": [  
    {  
      "name": string,  
      "policy": string,  
      #"allow_network": [string],  
      #"deny_network": [string],  
      #"allow_header": [string],  
      #"deny_header": [string],  
      #"method_whitelist": [string],  
      #"method_blacklist": [string]  
    }  
  ]  
}
```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the ACCESS section configuration items.

## access\_list

A list of access control configuration objects for the ACCESS section.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### access\_list/name (Required)

Sets the ACCESS section name.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### access\_list/policy (Required)

Sets the order in which allow/deny rules are applied to requests.

| Item      | Description                               |
|-----------|-------------------------------------------|
| Data Type | String                                    |
| Values    | "blacklist"   "whitelist"   "method_only" |

### access\_list/allow\_network

Sets the IP addresses or networks/netmasks that are allowed for requests. The special value 'Allow = "all"' means all IP addresses.

| Item      | Description                             |
|-----------|-----------------------------------------|
| Data Type | Array (string)                          |
| Range     | Up to 256 items (within 255 characters) |

### access\_list/deny\_network

Sets the IP addresses or networks/netmasks that are denied for requests.

| Item      | Description                             |
|-----------|-----------------------------------------|
| Data Type | Array (string)                          |
| Range     | Up to 256 items (within 255 characters) |

## access\_list/allow\_header

Sets the headers that are allowed in the request. The value of the <header field name> in the request matches the pattern of <regular expression>.

| Item      | Description                              |
|-----------|------------------------------------------|
| Data Type | Array (string)                           |
| Range     | Up to 256 items (within 255 characters)  |
| Format    | <header field name> <regular expression> |

## access\_list/deny\_header

Sets the headers that are denied in the request. The value of the <header field name> in the request matches the pattern of <regular expression>.

| Item      | Description                              |
|-----------|------------------------------------------|
| Data Type | Array (string)                           |
| Range     | Up to 256 items (within 255 characters)  |
| Format    | <header field name> <regular expression> |

## access\_list/method\_whitelist

Sets the HTTP method to use. However, this cannot be set simultaneously with method\_blacklist.

| Item      | Description                                                                                                                                        |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type | Array (string)                                                                                                                                     |
| Values    | "GET", "POST", "PUT", "HEAD", "DELETE", "CONNECT", "OPTIONS", "TRACE", "PATCH", "PROPFIND", "PROPPATCH", "MKCOL", "COPY", "MOVE", "LOCK", "UNLOCK" |

## access\_list/method\_blacklist

Sets HTTP methods to be excluded. However, this cannot be set simultaneously with method\_whitelist.

| Item      | Description                                                                                                                                        |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type | Array (string)                                                                                                                                     |
| Values    | "GET", "POST", "PUT", "HEAD", "DELETE", "CONNECT", "OPTIONS", "TRACE", "PATCH", "PROPFIND", "PROPPATCH", "MKCOL", "COPY", "MOVE", "LOCK", "UNLOCK" |

### 3.2.2. Default Settings

The "#default\_server\_access" configuration is internally created in the ACCESS section. This configuration is referenced as the default access setting in /server/http/common\_config and its subordinate common\_config sections.

If the "#default\_server\_access" setting is omitted, the default settings are applied to the access\_list as follows.

```
{
  "access": {
    "access_list": [
      {
        "name": "#default_server_access",
        "policy": "method_only",
        "method_whitelist": [ "GET", "POST", "HEAD", "OPTIONS" ]
      }
    ]
  }
}
```

Specifying the "#default\_server\_access" setting can change the rules applied when access settings are omitted in /server/http/common\_config and its subordinate common\_config sections.

### 3.2.3. Example

The following is an example configuration of the ACCESS section.

```
{
  "access": {
    "access_list": [
      {
        "name": "access1",
        "policy": "blacklist",
        "deny_network": [ "192.168.1.43/255.255.255.0" ]
      }
    ]
  }
}
```

## 3.3. ALIAS Section

The ALIAS section maps a URI to a physical directory path on the server regardless of the document root. Aliasing can map a directory regardless of the Document Root, which is convenient from a management perspective.

### 3.3.1. Configuration Items

The following is the environment configuration format of the ALIAS section.

```
#"alias": {
  #"alias_list": [
    {
      "name": string,
      "url": string,
      "real_path": string      # $ENV
    }
  ]
}
```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the ALIAS section configuration items.

#### alias\_list

A list of ALIAS configuration.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

#### alias\_list/name (Required)

The name of the ALIAS configuration. This 'name' must be set when using the functionality of the ALIAS section in other sections.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

#### alias\_list/url (Required)

Sets the URI to be specified as an alias. Matching is checked by comparing the request URL with the configured URI, starting from the first byte, up to the length of the URI.

| Item      | Description                  |
|-----------|------------------------------|
| Data Type | String                       |
| Range     | Up to 255 characters         |
| Format    | Must start with a slash (/). |

### alias\_list/real\_path (Required)

Sets the physical directory path on the server. Environment variables can be used during configuration. Relative paths that do not start with '/' are automatically replaced with "\$WEBTOB\_HOME/relative path".

| Item      | Description                |
|-----------|----------------------------|
| Data Type | String                     |
| Range     | Up to 255 characters       |
| Format    | Must end with a slash (/). |

### 3.3.2. Example

The following is an example configuration of the ALIAS section.

```
{
  "alias": {
    "alias_list": [
      {
        "name" : "alias1",
        "url": "/external/path/",
        "real_path": "/internal/real/path/"
      }
    ]
  }
}
```

## 3.4. DESTINATION Section

The DESTINATION section configures the backend server to be used for processing requests and receiving responses. It must include one of **'jeus'**, **'reverse proxy'**, or **'htmls'**.

### 3.4.1. Configuration Items

The following is the environment configuration format of the DESTINATION section.

```
"destination": {
  #"jeus": [
    {
      "name": string,
      #"server_schedule": string,                # "RR"
      #"connection_schedule": string,            # "RR"
      #"enable_flexible_sticky_session_routing": boolean,  # false
      #"enable_request_level_ping": boolean,        # false
      #"request_level_ping_timeout": integer        # 3 (0-INT_MAX)
      #"request_level_ping_retry_count": integer    # 0 (0-INT_MAX)
    }
  ]
}
```

```

    # "headers": [string],
    # "session_id_cookie_key": string, # "JSESSIONID"
    # "rewrite_cookie_domain": string,
    # "rewrite_cookie_path": {
        "from": string,
        "to": string
    },
    # "enable_cache": boolean, # false
    # "cache_refresh": integer, # 3600 (1-INT_MAX)
    # "max_queue_count": integer, # 0 (0-INT_MAX)
    # "max_queue_url": string,
    # "max_queue_status_code": integer, # 503
    # "queue_timeout": integer, # 0 (0-INT_MAX)
    # "backup_server": string
},
],
# "reverse_proxy": {
    # "common_config": { # COMMON
        # "headers": [string],
        # "compression": [string],
        # "compression_min_size": integer, # 0 (0-INT_MAX)
        # "rewrite_redirect": [
            {
                "original_uri": string,
                "redirect_path": string
            }
        ],
        # "rewrite_cookie_domain": string,
        # "rewrite_cookie_path": {
            "from": string,
            "to": string,
        },
        # "max_queue_count": integer, # 0 (0-INT_MAX)
        # "max_queue_url": string,
        # "max_queue_status_code": integer, # 503
        # "websocket_connections_max": integer, # 0 (0-INT_MAX)
        # "websocket_session_timeout": integer, # 0 (0-INT_MAX)
        # "set_host_header": string,
        # "queue_timeout": integer, # 0 (0-INT_MAX)
        # "name_resolution_interval": integer, # 0 (0-INT_MAX)
        # "server_health_check": {
            # "retry_count": integer, # 3 (0-INT_MAX)
            # "failback_interval": integer, # 60 (0-INT_MAX)
            # "http_failure": [string], # []
            # "connection_timeout": integer, # 5 (0-INT_MAX)
            # "enable_name_resolution_on_fail": boolean # true
        },
        # "rewrite_html_url": [
            {
                "tag": string,
                "attribute": [string],
                "from": string,
                "to": string
            }
        ],
        # "rewrite_html_max_size": integer # 10240 (1-INT_MAX)
    },
    # "reverse_proxy_group": [
        {

```

```

        "name": string,
        "reverse_proxy_server": [
            {
                "name": string,
                "address": string,
                #"enable_proxy_ssl": boolean,                # false
                #"proxy_ssl_name": [string],
                #"persistent_server_connections_min": integer, # 0 (0-INT_MAX)
                #"persistent_server_connections_max": integer, # 0 (0-INT_MAX)
                #"persistent_server_check_time": integer,      # 30 (0-INT_MAX)
                #"persistent_server_check_url": string,
                #"persistent_server_timeout": integer,         # 300 (0-INT_MAX)
                #"enable_cache": boolean,                      # false
                #"cache_refresh": integer,                    # 3600 (1-INT_MAX)
                #"sticky_session_routing_id": string,
                #"is_backup_server": boolean,                 # false
                #"load_balancing_factor": integer,            # 1 (1-INT_MAX)
                #"common_config": {...}                      # COMMON
            }
        ],
        #"server_schedule": string,                          # "RR"
        #"sticky_session_routing": {
            #"policy": string,                                # "UseOriginalCookie"
            #"session_id_cookie_key": string,                 # "JSESSIONID"
            #"enable_flexible_sticky_session_routing": boolean # false
        },
        #"common_config": {...}                              # COMMON
    }
]
},
#"htmls": [
    {
        "name": string,
        #"headers": [string],
        #"compression": [string],
        #"compression_min_size": integer,                    # 0 (0-INT_MAX)
        #"enable_cache": boolean,                            # true
        #"cache_refresh": integer,                           # 3600 (1-INT_MAX)
        #"queue_timeout": integer,                           # 0 (0-INT_MAX)
        #"enable_sendfile": boolean,                         # false
        #"sendfile_min_size": integer,                       # 0 (0-INT_MAX)
        #"enable_etag": boolean                              # true
    }
],
#"fcgi": {
    #"common_config": {                                     # COMMON
        #"headers": [string],
        #"compression": [string],
        #"compression_min_size": integer,                   # 0 (0-INT_MAX)
        #"max_queue_count": integer,                        # 0 (0-INT_MAX)
        #"max_queue_url": string,
        #"max_queue_status_code": integer,                  # 503
        #"queue_timeout": integer,                          # 0 (0-INT_MAX)
        #"connection_timeout": integer,                     # 5 (0-INT_MAX)
        #"name_resolution_interval": integer,               # 0 (0-INT_MAX)
    },
    "fcgi_group": [
        {
            "name": string,

```

```

    "fcgi_server": [
      {
        "name": string,
        "address": string,
        #"param": [
          {
            "key": string,
            "value": string
          }
        ],
        #"enable_cache": boolean,           # false
        #"cache_refresh": integer,         # 3600 (1-INT_MAX)
        #"is_backup_server": boolean,      # false
        #"load_balancing_factor": integer, # 1 (1-INT_MAX)
        #"common_config": {...}           # COMMON
      }
    ],
    #"server_schedule": string,           # "RR"
    #"common_config": {...}              # COMMON
  },
]
}

```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the DESTINATION section configuration items.

## jeus

Set this when performing services using WJP by integrating WebtoB with JEUS.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

## jeus/name (Required)

Sets the name of JEUS. The specified name must match the name set in wjp/wjp\_servers/name.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

## jeus/server\_schedule

Sets the method for specifying a server to process requests when multiple servers are defined in wjp/wjp\_servers.

| Item          | Description |
|---------------|-------------|
| Data Type     | String      |
| Range         | "RR"        |
| Default Value | "RR"        |

The following describes each configuration value.

| Value | Description                                                         |
|-------|---------------------------------------------------------------------|
| RR    | Assigns requests sequentially to JEUS using the round robin method. |

### **jeus/connection\_schedule**

Sets the method for specifying a connection to process requests when multiple connections are established to the JEUS server.

| Item          | Description |
|---------------|-------------|
| Data Type     | String      |
| Range         | "RR"        |
| Default Value | "RR"        |

The following describes each configuration value.

| Value | Description                                                                |
|-------|----------------------------------------------------------------------------|
| RR    | Assigns requests to connections sequentially using the round robin method. |

### **jeus/enable\_flexible\_sticky\_session\_routing**

Sets whether to enable flexible routing during session routing.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following describes each configuration value.

| Value | Description                                                                                                                                                                                                                                                                                    |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| True  | If connections up to persistent_server_connections_max of the internal server are in the RUN state when using the sticky_session_routing_id for session routing, requests are routed to another internal server connection with a different sticky_session_routing_id instead of being queued. |

| Value | Description                                                                                                                                                                                                                                           |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| False | Uses the sticky session routing by default. Requests are routed only to internal server connections with the same sticky_session_routing_id, and requests are queued if connections up to persistent_server_connections_max are all in the RUN state. |



If you are using flexible routing, requests from different clients with the same JSESSIONID may be routed to different servers. **(Default setting is recommended.)**

## jeus/enable\_request\_level\_ping

Sets whether to send a PING to the server before forwarding a request. This setting is useful when multiple JEUS servers are connected and a server may have difficulty responding to requests due to Out Of Memory (OOM).

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following describes each configuration value.

| Value | Description                                                                                                          |
|-------|----------------------------------------------------------------------------------------------------------------------|
| True  | Sends a PING for every request forwarded to the JEUS server and only forwards the request if a response is received. |
| False | Directly forwards the request without sending a PING for all requests forwarded to the JEUS server.                  |



This may result in performance degradation as it requires sending a PING and waiting for a response for every request. **(Default setting is recommended.)**

## jeus/request\_level\_ping\_timeout

Sets the time to wait for a response after sending a PING. If no response is received within the specified time, the connection is terminated, and another JEUS server is rescheduled to send a PING.

| Item      | Description |
|-----------|-------------|
| Data Type | Integer     |
| Unit      | seconds     |
| Range     | 0 ~ INT_MAX |

| Item          | Description |
|---------------|-------------|
| Default Value | 3           |

### **jeus/request\_level\_ping\_retry\_count**

Sets the number of retry attempts if sending a PING to all Jengines within the JEUS server fails.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **jeus/headers**

Sets the name of the HEADERS section to apply.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

### **jeus/session\_id\_cookie\_key**

Sets the Key name of the HTTP cookie used for session routing.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "JSESSIONID"         |

### **jeus/rewrite\_cookie\_domain**

Modifies the "domain=" value of the Cookie header field of the response from the internal server. If the domain string specified in "domain=" matches the string specified in RewriteCookieDomain, the "domain=" value is replaced with the domain of the user request.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "JSESSIONID"         |

Using the following configuration, if the WebtoB domain is webtob, the cookie of the original response "jsessionid=abc, domain=internal.server.com" is changed to "jsessionid=abc, domain=webtob".

```
"rewrite_cookie_domain": "internal.server.com"
```

## jeus/rewrite\_cookie\_path

Modifies the "path=" value of the cookie header field of the internal server response.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## jeus/rewrite\_cookie\_path/from

Specifies the string to be modified in the "path=" value.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## jeus/rewrite\_cookie\_path/to

Specifies the string to replace the "path=" value.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## jeus/enable\_cache

Sets whether the content is cached.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

## jeus/cache\_refresh

Sets the value for calculating the validity period of cached responses when caching responses received from JEUS.

When WebtoB responds to a client's request with cached content, it checks whether the cached content is valid. The validity is determined based on the set value (valid time).

| Item          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type     | Integer                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Unit          | seconds                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Range         | 1 ~ INT_MAX                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Default Value | 3600                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Priority      | The priority applied to this setting is as follows:<br><br>1. Cache-Control:max-age: If the Cache-Control:max-age value is in the Response Header when the header is being cached, the header is specified to be valid for the max-age value (second).<br><br>2. Expires: If the Expires value is in the Response Header when the header is being cached, the header is specified to be valid until the Expires value (hour).<br><br>3. cache_refresh: If there are no 'Cache-Control:max-age' or 'Expires' values in the response header, the response received from JEUS is valid for cache_refresh (in seconds) after being cached. |

### jeus/max\_queue\_count

When a surge in client requests overloads a server and the server is no longer able to respond to new requests, it is necessary to ignore them.

Once the number of client requests in the queue reaches a certain limit, any further incoming requests will not be queued. Instead, the server will respond to the client with an error.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### jeus/max\_queue\_url

Sets the page to be served instead when the server queue is full.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## jeus/max\_queue\_status\_code

Sets the status code to be returned when the server queue is full.

| Item          | Description                             |
|---------------|-----------------------------------------|
| Data Type     | String                                  |
| Range         | 301   302   303   307   308   410   503 |
| Default Value | 503                                     |

## jeus/queue\_timeout

Specifies the timeout for user requests in a server queue.

When a request cannot be processed due to a large volume of user requests, the request waits in the server queue until a server process becomes available.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value            | Description                                                                                                                |
|------------------|----------------------------------------------------------------------------------------------------------------------------|
| 0                | There is no time limit for waiting in the server queue.                                                                    |
| Positive Integer | Requests waiting longer than the specified time will be removed from the queue and respond with "503 Service Unavailable". |

The following is an example of how to configure max\_queue:

```
...
"destination": {
  "jeus": {
    "max_queue_count": 3,
    "max_queue_url": "/jsvtest/common/test.html",
    "max_queue_status_code": 302,
    "queue_timeout": 100
  }
},
...
```

With the above settings, when there are 2 JEUS connections and 10 calls are made, the response will include 2 page calls, 3 queue page calls, and 5 calls to the max\_queue\_url.

## jeus/backup\_server

Sets the server to be used as a backup when all servers are not ready.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

## jeus/enable\_block\_listen

If the server goes down, the service ports associated with the virtual hosts are closed to block any new client connections. When the server is restored, the ports reopen for new connections.

If a node fails in an L4 environment composed of multiple nodes, its service port stops listening. The L4 load balancer distributes client connections to nodes that are operating normally.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |



If the service port is a well-known port, the block-listen feature is not supported.

## reverse\_proxy

This is a type of HTTP proxy used to forward requests to internal servers and can be used for web application server (WAS) integration.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## reverse\_proxy/common\_config

A common setting for reverse proxies. If set in the parent item, the settings will be reflected to child items without additional configuration.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

| Item     | Description                                                                                                                                                                                    |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Priority | <p>The priority applied to this setting is as follows:</p> <ol style="list-style-type: none"> <li>1. reverse_proxy_server</li> <li>2. reverse_proxy_group</li> <li>3. reverse_proxy</li> </ol> |

## reverse\_proxy/common\_config/headers

Sets the name of the HEADERS section to apply.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

## reverse\_proxy/common\_config/compression

Sets the target for response compression. Specifies the MIME-type (content-type of the response) to be compressed. The target response is compressed with GZIP, then sent to the client.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 32 items (within 255 characters) |



Compression can reduce the network traffic, but it may degrade server performance.

Files with low compression ratio, such as zip or jpeg, should not be compressed if possible to avoid server overhead.



Compression can only be used for requests with Accept-Encoding set to GZIP or deflate in the HTTP request header.

## reverse\_proxy/common\_config/compression\_min\_size

Specifies the minimum size of the response to compress.

If the value of the Content-Length response header is greater than the specified size, the response is compressed. However, this does not apply to chunked responses, as it is difficult to determine the size of the response.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### reverse\_proxy/common\_config/rewrite\_redirect

Modifies the values in the "Location" and "Content-Location" headers in the response from the internal server. If that field starts with "original\_uri", replace the part that matches "original\_uri" with "redirect\_path".

| Item      | Description                                                                        |
|-----------|------------------------------------------------------------------------------------|
| Data Type | Array (object)                                                                     |
| Range     | Up to 16 items (each original_uri and redirect_path must be within 255 characters) |

In the following example, if the Location of the original response is "http://internal.server.com:80/docs\_kr/abc.html", it will be changed to "/internal\_kr/abc.html".

```
"rewrite_redirect": [
  {
    "original_uri": "http://internal.server.com:80/docs/",
    "redirect_path": "/internal/"
  },
  {
    "original_uri": "http://internal.server.com:80/docs_kr",
    "redirect_path": "/internal_kr/"
  },
  {
    "original_uri": "http://internal.server.com:80/docs_ch/",
    "redirect_path": "/internal_ch/"
  }
]
```

### reverse\_proxy/common\_config/rewrite\_cookie\_domain

Modifies the "domain=" value of the Cookie header field of the response from the internal server. If the domain string specified in "domain=" matches the string specified in rewrite\_cookie\_domain, the "domain=" value is replaced with the domain of the user request.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | 0                    |

Using the following configuration, if the WebtoB domain is webtob, the cookie of the original response "jsessionid=abc, domain=internal.server.com" is changed to "jsessionid=abc, domain=webtob".

```
"rewrite_cookie_domain": "internal.server.com"
```

### reverse\_proxy/common\_config/rewrite\_cookie\_path

Modifies the "path=" value of the cookie header field of the internal server response.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

Using the following configuration, the original response cookie "jsessionid=abc, path=/jeus/application" is changed to "jsessionid=abc, path=/jeus\_proxy/application".

```
"rewrite_cookie_path": {  
  "from": "jeus",  
  "to": "/jeus_proxy"  
}
```

### reverse\_proxy/common\_config/rewrite\_cookie\_path/from (Required)

Specifies the string to be modified in the "path=" value.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### reverse\_proxy/common\_config/rewrite\_cookie\_path/to (Required)

Specifies the string to replace the "path=" value.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### reverse\_proxy/common\_config/max\_queue\_count

When a surge in client requests overloads a server and the server is no longer able to respond to new requests, it is necessary to ignore them.

Once the number of client requests in the queue reaches a certain limit, any further incoming requests will not be queued. Instead, the server will respond to the client with an error.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **reverse\_proxy/common\_config/max\_queue\_url**

Sets the page to be served instead when the server queue is full.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **reverse\_proxy/common\_config/max\_queue\_status\_code**

Sets the status code to be returned when the server queue is full.

| Item          | Description                             |
|---------------|-----------------------------------------|
| Data Type     | Integer                                 |
| Range         | 301   302   303   307   308   410   503 |
| Default Value | 503                                     |

### **reverse\_proxy/common\_config/websocket\_connections\_max**

Specifies the maximum number of connections when connection is upgraded from HTTP to WebSocket protocol.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                                             |
|-------|---------------------------------------------------------|
| 0     | No limit is set on the number of WebSocket connections. |

## reverse\_proxy/common\_config/websocket\_session\_timeout

Specifies the timeout for WebSocket connection.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                     |
|-------|---------------------------------|
| 0     | Timeout check is not performed. |



It is recommended to set this value based on the timeout setting of the WebSocket session on the internal server.

## reverse\_proxy/common\_config/set\_host\_header

Specifies the Host header when forwarding the request to the internal server using reverse proxy.

If not set, the value specified in ServerAddress is used. If set to "\$BypassHostHeader", the client Host header is used without modification.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | 0                    |

## reverse\_proxy/common\_config/queue\_timeout

Specifies the timeout for user requests in a server queue.

When a request cannot be processed due to a large volume of user requests, the request waits in the server queue until a server process becomes available.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value            | Description                                                                                                                |
|------------------|----------------------------------------------------------------------------------------------------------------------------|
| 0                | There is no time limit for waiting in the server queue.                                                                    |
| Positive Integer | Requests waiting longer than the specified time will be removed from the queue and respond with "503 Service Unavailable". |

### **reverse\_proxy/common\_config/name\_resolution\_interval**

Sets the frequency at which Hostname Resolution is performed.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                                                 |
|-------|-------------------------------------------------------------|
| 0     | Follows the operating system's Time-To-Live (TTL) settings. |

### **reverse\_proxy/common\_config/server\_health\_check**

Configures health check information for reverse proxy groups.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

The following is an example of using the server\_health\_check:

```
"destination": {
  "reverse_proxy ": {
    "common_config": {
      "server_health_check":{
        "retry_count":4,
        "failback_interval":5,
        "http_failure":["http_invalid", "http_4xx"],
        "connection_timeout": 5,
        "enable_name_resolution_on_fail":false
      }
    }
  }
}
```

### **reverse\_proxy/common\_config/server\_health\_check/retry\_count**

Sets the number of retry attempts to determine failover.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 3           |

### **reverse\_proxy/common\_config/server\_health\_check/failback\_interval**

Sets the health check interval to attempt a failback after a failover.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 60          |

The following describes each configuration value.

| Value | Description                            |
|-------|----------------------------------------|
| 0     | Does not perform failover or failback. |

### **reverse\_proxy/common\_config/server\_health\_check/http\_failure**

Sets the criteria for determining HTTP response failures.

| Item            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type       | Array (string)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Allowed Options | The following options can treat certain types of HTTP responses as failures: <ul style="list-style-type: none"><li>◦ http_invalid: Considers an invalid HTTP message as a failure.</li><li>◦ http_4xx: Considers 400 HTTP responses as failures.</li><li>◦ http_5xx: Considers 500 HTTP responses as failures.</li><li>◦ http_{3-digit number}: Considers HTTP responses corresponding to the specified {3-digit number} as failures. Allowed numbers are 403, 404, 429, 500, 502, 503, and 504.</li></ul> |

### **reverse\_proxy/common\_config/server\_health\_check/connection\_timeout**

Specifies the retry interval for TCP connection with the internal server.

| Item          | Description           |
|---------------|-----------------------|
| Data Type     | Integer               |
| Range         | 0 ~ INT_MAX (seconds) |
| Default Value | 5 (seconds)           |

### **reverse\_proxy/common\_config/server\_health\_check/enable\_name\_resolution\_on\_fail**

Sets whether to attempt Hostname Resolution again when a failure is determined.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | True        |

### **reverse\_proxy/common\_config/rewrite\_html\_url**

Used to replace the URL included in the HTML page response body. If the URL includes a host, the host is replaced with the WebtoB server address used in the request.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |
| Range     | Up to 64 items |

The following is an example of changing the URL from http://test2:80 to /proxy/ for the src, longdesc, and usemap attributes of the img tag.

```
"rewrite_html_url": [
  {
    "tag": "img",
    "attribute": ["src", "longdesc"],
    "from": "http://test2:80",
    "to": "/proxy/"
  }
]
```

### **reverse\_proxy/common\_config/rewrite\_html\_url/tag (Required)**

Specifies the tag name.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## **reverse\_proxy/common\_config/rewrite\_html\_url/attribute (Required)**

Specifies the attribute name.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 64 items (within 255 characters) |

## **reverse\_proxy/common\_config/rewrite\_html\_url/from (Required)**

Specifies the string to be modified in the URL.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## **reverse\_proxy/common\_config/rewrite\_html\_url/to (Required)**

Specifies the string to which the URL will be modified.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## **reverse\_proxy/common\_config/rewrite\_html\_max\_size**

If the response is an HTML page (Content-Type: text/html), specific tag values inside the page can be modified. This value sets the maximum internal buffer size that can be used. If the response size is greater than the specified value, the original response is sent to the client.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 10240       |

## **reverse\_proxy/reverse\_proxy\_group (Required)**

You can manage the reverse\_proxy settings as a group and configure multiple servers.

By setting up multiple reverse proxy (internal) servers, you can group them for load balancing and sticky session routing, and integrate them with Web Application Server (WAS).

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### **reverse\_proxy/reverse\_proxy\_group/name (Required)**

Specifies the reverse\_proxy\_group name.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server (Required)**

You can manage different configurations for each ip:port in one reverse\_proxy\_group.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |
| Range     | Up to 32 items |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/name (Required)**

Specifies the name of the reverse proxy server.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/address (Required)**

Specifies the internal server address to which the request is sent.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/enable\_proxy\_ssl**

Specifies whether to use SSL/TLS to connect to the internal server. If set, the PROXY\_SSL section item can be set in proxy\_ssl\_name.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/proxy\_ssl\_name**

Specifies the name of the ssl/proxy\_ssl\_configs section to be used. This setting is applied only when enable\_proxy\_ssl is set to 'true'.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 100 items (within 31 characters) |

The following is an example of using proxy\_ssl\_name:

```
"destination": {
  "reverse_proxy":{
    "reverse_proxy_server":[
      {
        "address":"internal.server.com:80",
        "name":"rproxy1",
        "enable_proxy_ssl": true,
        "proxy_ssl_name":["ssl1"]
      }
    ]
  }
}
```

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/persistent\_server\_connections\_min**

Specifies the minimum number of connections required to maintain connection with an internal server after request processing is complete.

A new connection with an internal server is created when there is a new request, and as long as the internal server doesn't terminate the connection it is maintained and reused for subsequent requests.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/persistent\_server\_connections\_max**

Specifies the maximum number of connections required to maintain connection with an internal server after request processing is complete.

When there is a new request, a new connection is created if all connections are currently processing other requests and the request is queued if the number of connections has reached the specified maximum limit.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/persistent\_server\_check\_url**

Specifies to internally use the HTTP HEAD request as the ping message for maintaining internal server connections.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "/"                  |

Connections are maintained by using a ping message to check for the connection state after sending an HTTP request to the internal server and receiving an HTTP response. If the response message is not 200, the connection is terminated. An application must be implemented on the internal server to respond to the ping.

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/persistent\_server\_check\_time**

Specifies the interval for checking connection with an internal server in order to manage internal server connections.

To maintain internal server connections, this must be set to a value less than the keepalive\_timeout of the internal server. A single ping is sent to connections in the Ready state, and the connection is disconnected if no reply is received within the persistent\_server\_check\_time period.

| Item      | Description |
|-----------|-------------|
| Data Type | Integer     |
| Unit      | seconds     |

| Item          | Description |
|---------------|-------------|
| Range         | 0 ~ INT_MAX |
| Default Value | 30          |

The following describes each configuration value.

| Value | Description                                                                                             |
|-------|---------------------------------------------------------------------------------------------------------|
| 0     | no ping is sent and connections are maintained for the keepalive_timeout period of the external server. |

## **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/persistent\_server\_timeout**

Specifies the timeout for terminating a connection in the Ready state when the number of internal server connections has reached the persistent\_server\_connections\_min.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 0 ~ INT_MAX |
| Default Value | 300         |

The following describes each configuration value.

| Value | Description                            |
|-------|----------------------------------------|
| 0     | No timeout check for idle connections. |

The following is an example of using persistent\_server:

```
"destination": {
  "reverse_proxy":{
    "reverse_proxy_group":[
      {
        "name": "rproxyg1",
        ...
        "persistent_server_connections_min":10,
        "persistent_server_connections_max":15,
        "persistent_server_check_time": 25,
        "persistent_server_check_url":"/jsvtest/common/test.html",
        "persistent_server_timeout":3000
      }
    ]
  }
}
```

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/enable\_cache**

Sets whether the content is cached.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/cache\_refresh**

Specifies the value to calculate the time a cached response is valid when a response handled by a reverse proxy is cached.

When WebtoB responds to a client's request with cached content, it checks whether the cached content is valid. The validity is determined based on the set value (valid time).

| Item          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type     | Integer                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Unit          | seconds                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Range         | 1 ~ INT_MAX                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Default Value | 3600                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Priority      | <p>The priority applied to this setting is as follows:</p> <ol style="list-style-type: none"><li>1. Cache-Control:max-age: If the Cache-Control:max-age value is in the Response Header when the header is being cached, the header is specified to be valid for the max-age value (second).</li><li>2. Expires: If the Expires value is in the Response Header when the header is being cached, the header is specified to be valid until the Expires value (hour).</li><li>3. cache_refresh: If there are no 'Cache-Control:max-age' or 'Expires' values in the response header, the response received from JEUS is valid for cache_refresh (in seconds) after being cached.</li></ol> |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/sticky\_session\_routing\_id**

Used for integration with a specific internal server (web application server) and using sticky session routing.

Use the name of the engine that corresponds to the Sticky Session id (JSESSIONID) value set in the Set-Cookie response header by the internal server (WAS). For example, if the "JSESSIONID" value in the Set-Cookie response header is "PI1xfBkEVbUu2cj20CUNIHJoWLMU.xxx\_servlet\_engine1", use the value following the delimiter (period) which is "xxx\_servlet\_engine1".

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/is\_backup\_server**

Sets whether this server should be used as a backup server in case all other reverse proxy servers are not ready.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/load\_balancing\_factor**

Sets the ratio at which requests will be distributed to the corresponding reverse proxy servers in the reverse proxy group.

Requests are distributed in the ratio of "(load\_balancing\_factor of the reverse proxy server) / (sum of load\_balancing\_factors of all reverse proxy servers in the group)".

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 1           |

### **reverse\_proxy/reverse\_proxy\_group/reverse\_proxy\_server/common\_config**

A common setting for reverse proxies. If set in the parent item, the settings will be reflected to child items without additional configuration.

| Item      | Description                                                                                                                                                                                    |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                                                                                                                         |
| Priority  | <p>The priority applied to this setting is as follows:</p> <ol style="list-style-type: none"> <li>1. reverse_proxy_server</li> <li>2. reverse_proxy_group</li> <li>3. reverse_proxy</li> </ol> |

## reverse\_proxy/reverse\_proxy\_group/server\_schedule

Sets the method for specifying the reverse proxy to process requests when multiple reverse proxies are set up.

| Item          | Description |
|---------------|-------------|
| Data Type     | String      |
| Range         | "RR"        |
| Default Value | "RR"        |

The following describes each configuration value.

| Value | Description                                                                    |
|-------|--------------------------------------------------------------------------------|
| RR    | Assigns requests to reverse proxies sequentially using the round robin method. |

## reverse\_proxy/reverse\_proxy\_group/sticky\_session\_routing

Settings related to session routing.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## reverse\_proxy/reverse\_proxy\_group/sticky\_session\_routing/policy

Sets the session routing policy.

| Item          | Description                                                   |
|---------------|---------------------------------------------------------------|
| Data Type     | String                                                        |
| Range         | "AddNewCookie"   "ModifyOriginalCookie"   "UseOriginalCookie" |
| Default Value | "UseOriginalCookie"                                           |

## reverse\_proxy/reverse\_proxy\_group/sticky\_session\_routing/session\_id\_cookie\_key

Specifies the Key name of the HTTP cookie used for session routing.

| Item          | Description  |
|---------------|--------------|
| Data Type     | String       |
| Default Value | "JSESSIONID" |

## reverse\_proxy/reverse\_proxy\_group/sticky\_session\_routing/enable\_flexible\_sticky\_session\_routing

Sets whether to enable flexible routing during session routing.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following describes each configuration value.

| Value | Description                                                                                                                                                                                                                                                                                    |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| True  | If connections up to persistent_server_connections_max of the internal server are in the RUN state when using the sticky_session_routing_id for session routing, requests are routed to another internal server connection with a different sticky_session_routing_id instead of being queued. |
| False | Uses the sticky session routing by default. Requests are routed only to internal server connections with the same sticky_session_routing_id, and requests are queued if connections up to persistent_server_connections_max are all in the RUN state.                                          |



If you are using flexible routing, requests from different clients with the same JSESSIONID may be routed to different servers. (Default setting is recommended.)

## reverse\_proxy/reverse\_proxy\_group/common\_config

A common setting for reverse proxies. If set in the parent item, the settings will be reflected to child items without additional configuration.

| Item      | Description                                                                                                                      |
|-----------|----------------------------------------------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                                                           |
| Priority  | The priority applied to this setting is as follows:<br><br>1. reverse_proxy_server<br>2. reverse_proxy_group<br>3. reverse_proxy |

## fcgi

Used to forward HTTP requests to an internal server (example: php-fpm) using the FastCGI (Common Gateway Interface) protocol.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## fcgi/common\_config

Common FastCGI settings. If configured in a parent item, the settings are applied to child items without needing to be set again.

| Item      | Description                                                                                                                                                  |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                                                                                       |
| Priority  | The priority applied to this setting is as follows: <ol style="list-style-type: none"> <li>1. fcgi_server</li> <li>2. fcgi_group</li> <li>3. fcgi</li> </ol> |

## fcgi/common\_config/headers

Sets the name of the HEADERS section to apply.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

## fcgi/common\_config/compression

Specifies the targets for response compression. The MIME types to compress are set, and the response's Content-Type is used. Responses matching these types are compressed using GZIP before being sent to the client.



Compression is applicable only to HTTP requests whose Accept-Encoding header includes GZIP or deflate.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 32 items (within 255 characters) |



Compression can reduce network traffic, but it may degrade server performance. Since compressing files with low compression ratios (example: ZIP, JPEG) imposes

unnecessary load to the server, such MIME types must be excluded from compression.

### **fcgi/common\_config/compression\_min\_size**

Sets the minimum response size for compression.

Compresses the response if the Content-Length header value exceeds the set value. However, compression is not applied to chunked responses, as their size cannot be determined in advance.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **fcgi/common\_config/max\_queue\_count**

When client requests increase excessively and disrupt normal service, further requests need to be ignored.

When the number of client requests in the queue exceeds a certain threshold, newly arriving requests are not queued and the client receives an immediate error response.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **fcgi/common\_config/max\_queue\_url**

Sets the alternative page to be served when the server queue is full.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **fcgi/common\_config/max\_queue\_status\_code**

Sets the status code to be returned when the server queue is full.

| Item          | Description                             |
|---------------|-----------------------------------------|
| Data Type     | Integer                                 |
| Range         | 301   302   303   307   308   410   503 |
| Default Value | 503                                     |

### **fcgi/common\_config/queue\_timeout**

Sets the timeout for user requests waiting in the server queue.

If there is no server processes available to handle excessive user requests, the requests wait in the server queue until a server process becomes available to process them.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value            | Description                                                                                                                   |
|------------------|-------------------------------------------------------------------------------------------------------------------------------|
| 0                | No limit on the time requests can wait in the server queue                                                                    |
| Positive Integer | Requests that wait longer than the specified time are removed from the queue and receive a "503 Service Unavailable" message. |

### **fcgi/common\_config/name\_resolution\_interval**

Sets the interval at which hostname resolution is performed.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                                                 |
|-------|-------------------------------------------------------------|
| 0     | Follows the operating system's TTL (Time-To-Live) settings. |

### **fcgi/common\_config/connection\_timeout**

Sets the time period to wait for a response before retrying a TCP connection to the internal server.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 0 ~ INT_MAX |
| Default Value | 5           |

### **fcgi/fcgi\_group (Required)**

Manages fcgi settings in groups and allows configuring multiple servers.

Supports configuring multiple fcgi (internal) servers and grouping them for load balancing and sticky session routing. It can also be used for integration with a WAS.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### **fcgi/fcgi\_group/name (Required)**

Sets the name of the fcgi\_group.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### **fcgi/fcgi\_group/fcgi\_server (Required)**

Allows configuring different settings for each ip:port within a single fcgi\_group.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |
| Range     | Up to 32 items |

### **fcgi/fcgi\_group/fcgi\_server/name (Required)**

Sets the name of the fcgi server.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

## fcgi/fcgi\_group/fcgi\_server/address (Required)

Sets the address of the internal server to which requests are forwarded.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## fcgi/fcgi\_group/fcgi\_server/param

Sets the parameter to be passed to the fcgi server.



The following parameters are passed to the fcgi server by default without requiring separate configuration.

- CONTENT\_LENGTH, CONTENT\_TYPE
- DOCUMENT\_ROOT
- GATEWAY\_INTERFACE
- QUERY\_STRING
- REQUEST\_METHOD, REQUEST\_SCHEME, REQUEST\_URI
- REMOTE\_ADDR, REMOTE\_HOST, REMOTE\_PORT, REMOTE\_USER
- SCRIPT\_NAME
- SERVER\_ADDR, SERVER\_NAME, SERVER\_PORT, SERVER\_PROTOCOL, SERVER\_SOFTWARE
- SCRIPT\_FILENAME

Additionally, HTTP request headers are passed by default in the "HTTP\_{Header Key}: {Header Value}" format.



If you set a key identical to a default parameter, the existing value will be overwritten.

If the fcgi server is PHP-FPM and was built with the `--enable-force-cgi-redirect` option, the following configuration must be set.

```
"key": "REDIRECT_STATUS", "value": "200"
```

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### **fcgi/fcgi\_group/fcgi\_server/param/key**

Sets the key of the parameter to be passed to the fcgi server.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **fcgi/fcgi\_group/fcgi\_server/param/value**

Sets the value of the parameter to be passed to the fcgi server.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **fcgi/fcgi\_group/fcgi\_server/enable\_cache**

Sets whether to enable or disable content caching.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **fcgi/fcgi\_group/fcgi\_server/cache\_refresh**

Sets the value used to calculate the validity period of responses cached after being processed via fcgi.

When WebtoB responds to a client request using cached content, it checks whether the cached content is valid. Validity is determined based on the configured value (validity period).

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ INT_MAX |
| Default Value | 3600        |

| Item     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Priority | <p>The priority applied to this setting is as follows:</p> <ol style="list-style-type: none"> <li>1. Cache-Control:max-age: If the first response header to be cached includes a Cache-Control:max-age value, it is valid for max-age seconds.</li> <li>2. Expires: If the second response header to be cached includes an Expires value, it is valid until the time specified in Expires.</li> <li>3. cache_refresh: If the third response header does not include 'Cache-Control:max-age' and 'Expires', the cached content is valid for cache_refresh seconds after being cached from the response received from JEUS.</li> </ol> |

### fcgi/fcgi\_group/fcgi\_server/is\_backup\_server

Specifies whether the server should be used as a backup when all other internal servers are not ready.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### fcgi/fcgi\_group/fcgi\_server/load\_balancing\_factor

Sets the ratio of requests to be distributed to the fcgi server within the fcgi group.

Requests are distributed in the ratio of "(load\_balancing\_factor of the fcgi server) / (sum of load\_balancing\_factor of all fcgi servers in the group)."

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 1           |

### fcgi/fcgi\_group/fcgi\_server/common\_config

A common fcgi setting. If set in the parent item, the settings will be reflected to child items without additional configuration.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

| Item     | Description                                                                                                                                                         |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Priority | <p>The priority applied to this setting is as follows:</p> <ol style="list-style-type: none"> <li>1. fcgi_server</li> <li>2. fcgi_group</li> <li>3. fcgi</li> </ol> |

### fcgi/fcgi\_group/server\_schedule

Sets the method for specifying the fcgi server to process requests when multiple fcgi servers are set up.

| Item          | Description |
|---------------|-------------|
| Data Type     | String      |
| Range         | "RR"        |
| Default Value | "RR"        |

The following describes each configuration value.

| Value | Description                                                             |
|-------|-------------------------------------------------------------------------|
| RR    | Assigns requests sequentially to the fcgi using the round robin method. |

### fcgi/fcgi\_group/common\_config

A common fcgi setting. If set in the parent item, the settings will be reflected to child items without additional configuration.

| Item      | Description                                                                                                                                                         |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                                                                                              |
| Priority  | <p>The priority applied to this setting is as follows:</p> <ol style="list-style-type: none"> <li>1. fcgi_server</li> <li>2. fcgi_group</li> <li>3. fcgi</li> </ol> |

### htmls

Configures this setting to handle static file requests. Supported HTTP methods are "GET", "POST", and "HEAD", while a "405 Method Not Allowed" response is returned for all other methods.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### htmls/name (Required)

Specifies the name of the HTMLS.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### htmls/headers

Sets the name of the HEADERS section to apply.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

### htmls/compression

Sets the targets for compression in responses for static files. Specifies the MIME-types (content-types) of files to compress. The corresponding response will be compressed using GZIP before being sent to the client.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 32 items (within 255 characters) |



Compression can reduce the network traffic, but it may degrade server performance.

Files with low compression ratio, such as zip or jpeg, should not be compressed if possible to avoid server overhead.



Compression can only be used for requests with Accept-Encoding set to GZIP or deflate in the HTTP request header.

## htmls/compression\_min\_size

Specifies the minimum size of the response to compress. If the requested file size is greater than the specified size, the response is compressed.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

## htmls/enable\_cache

Sets whether the content is cached.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | True        |

## htmls/cache\_refresh

Sets the validity period for cached responses whose 'Content-Type' is 'text/html'. That is, a 'text/html' response will only be valid for the set number of seconds after being cached.

| Item          | Description |
|---------------|-------------|
| Data Type     | integer     |
| Unit          | seconds     |
| Range         | 1 ~ INT_MAX |
| Default Value | 3600        |



A conditional-GET request checks to see if the cached response has been modified. If it has been modified, the cache is deleted and updated with the new version.

## htmls/queue\_timeout

When a request cannot be processed due to a large volume of user requests, the request waits in the server queue until a server process becomes available.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value            | Description                                                                                                                |
|------------------|----------------------------------------------------------------------------------------------------------------------------|
| 0                | There is no time limit for waiting in the server queue.                                                                    |
| Positive Integer | Requests waiting longer than the specified time will be removed from the queue and respond with "503 Service Unavailable". |

### htmls/enable\_sendfile

Configures whether to use the sendfile function.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### htmls/sendfile\_min\_size

If a requested file size is greater than the specified value, the sendfile function is used.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ INT_MAX |
| Default Value | 8192        |

### htmls/enable\_etag

Configures whether to use ETag.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | True        |

The following describes each configuration value.

| Value | Description                                                              |
|-------|--------------------------------------------------------------------------|
| False | ETag is not added to HTTP response, and ETag of HTTP request is ignored. |

### 3.4.2. Default Settings

The "#default\_htmls" setting is created internally in HTMS configuration. It defines the HTMLS to which requests are routed if they do not match any rules in the SERVICE section.

If the "#default\_htmls" is omitted, the default settings are applied to htmls as follows.

```
{
  "destination": {
    "htmls": [
      {
        "name": "#default_htmls"
      }
    ]
  }
}
```

Specifying the "#default\_htmls" setting can change the HTMLS to which requests are routed when they do not match any rules registered in the SERVICE section.

### 3.4.3. Example

The following is an example of configuring the DESTINATION section:

```
"destination": {
  "jeus": [
    {
      "name": "MyGroup"
    }
  ],
  "reverse_proxy": {
    "common_config": {
      "server_health_check": {
        "retry_count": 3,
        "failback_interval": 5,
        "http_failure": ["http_invalid", "http_4xx"],
        "connection_timeout": 5
      }
    }
  },
  "reverse_proxy_group": [
    {
      "name": "rproxy1",
      "reverse_proxy_server": [
        {
          "address": "192.168.15.114:28080",
          "enable_proxy_ssl": false
        }
      ]
    }
  ]
}
```

```

    ]
  }
]
},
"fcgi": {
  "common_config": {
  },
  "fcgi_group": [
    {
      "name": "fcgi1",
      "fcgi_server": [
        {
          "address": "
        }
      ]
    }
  ]
}
],
"htmls": [
  {
    "name": "htmls1"
  }
]
}

```

## 3.5. DIRECTORY Section

The DIRECTORY section configures the properties of a specific directory.

It includes options such as **access** for restricting directory access, as well as **force\_mime\_type** and **default\_mime\_type** for setting file extensions within the directory.

### 3.5.1. Configuration Items

The following is the configuration format of the DIRECTORY section.

```

#"directory": {
  "directory_list": [
    {
      "name": string,
      "path": {
        "target": string,
        #"type": [string],           # "prefix"
      },
      #"target_http_servers": [string], # ["*"]
      #"access": [string],
      #"error_document": [string],
      #"force_mime_type": [string],
      #"default_mime_type": [string]
    }
  ]
}

```

```
}
```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the DIRECTORY section configuration items.

## directory\_list

A list of DIRECTORY settings.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### directory\_list/name (Required)

Sets the name of the DIRECTORY section.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### directory\_list/path (Required)

Sets the pathname of the directory to which the settings will be applied. The path can be an absolute path or a path relative to \$WEBTOB\_HOME.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

### directory\_list/path/target (Required)

Sets the pattern to match the directory pathname. If a match is found, the settings in the DIRECTORY section are applied to the request.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### directory\_list/path/type

Specifies the pattern type configured in the DIRECTORY section. The method used to match directory

paths varies depending on the pattern type.

| Item          | Description                |
|---------------|----------------------------|
| Data Type     | String                     |
| Range         | "prefix", "exact", "regex" |
| Default Value | "prefix"                   |

The following describes each configuration value.

| Value  | Description                                                                                                                                                                                                                                                                                                                |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| prefix | Matches when the pattern set in <code>directory/directory_list/path/target</code> is a prefix of the directory path.<br>(Example: The pattern <code>"/directory/"</code> matches paths such as <code>"/directory/a/"</code> , <code>"/directory/a/b/"</code> , and <code>"/directory/a/b/c/"</code> .)                     |
| exact  | Matches when the pattern set in <code>directory/directory_list/path/target</code> is exactly identical to the directory path.<br>(Example: The pattern <code>"/directory/"</code> matches only when the directory path is <code>"/directory/"</code> . It does not match any other directory paths.)                       |
| regex  | Matches when the directory path satisfies the regular expression pattern configured in <code>directory/directory_list/path/target</code> .<br>(Example: The pattern <code>"/directory.*"</code> matches directory paths such as <code>"/directory"</code> , <code>"/directory/"</code> , and <code>"/directory1"</code> .) |

### **directory\_list/target\_http\_servers**

Specifies the server name when the directory settings will apply only to a specific server. (Example: `"target_http_servers": "http1"`)

If the settings apply only to a specific vhost, set it to `"server name.vhost name"`. (Example: `"target_http_servers": "http1.vhost1"`)

| Item          | Description                           |
|---------------|---------------------------------------|
| Data Type     | Array (string)                        |
| Range         | Up to 64 items (within 31 characters) |
| Default Value | <code>["*"]</code>                    |

### **directory\_list/access**

Sets whether to allow a request based on its method, required headers, and the IP address it was sent from. The setting value must match the specified name set in `access/access_list/name`.

| Item      | Description |
|-----------|-------------|
| Data Type | String      |

| Item  | Description          |
|-------|----------------------|
| Range | Up to 255 characters |

### directory\_list/error\_document

Sets the name defined in the ERRORDOCUMENT section to replace the HTTP error page. This setting takes precedence over the error\_document setting in the SERVER section.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 64 items (within 255 characters) |

### directory\_list/force\_mime\_type

All resources within the specified directory are processed using the MIME-Type configured in force\_mime\_type.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 127 characters |

### directory\_list/default\_mime\_type

Specifies the default Content-Type for documents whose MIME-Type cannot be determined. This setting takes precedence over the default\_mime\_type setting in the SERVER section.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 127 characters |



Since the ext setting has a higher priority than the mime\_type setting, ext must be removed to apply the mime\_type settings.

## 3.5.2. Example

The following is an example of configuring the DIRECTORY section:

```
{
  "directory": {
    "directory_list": [
      {
```

```

        "name": "directory1",
        "path": {
            "type": "prefix",
            "target": "http/"
        },
        "access": "access1",
        "error_document": ["errorDoc1"],
        "force_mime_type": "text/html",
        "default_mime_type": "text/html"
    }
}
}
}

```

## 3.6. ERRORDOCUMENT Section

There are four ways to troubleshoot an error in WebtoB:

- Displays the error message defined in the source code.
- Displays the error message defined by the user.
- Redirect to a local URL.
- Redirect to an external URL.

For the second, third and fourth cases, the ERRORDOCUMENT section is set to redirect to a specific page for a particular HTTP response status code. All HTTP status codes, except HTTP 401 status code, can be configured.

### 3.6.1. Configuration Items

The following is the environment configuration format of the ERRORDOCUMENT section.

```

#"error_document": {
    #"error_document_list": [
        {
            "name": string,
            "status": integer,      # (HTTP status code)
            "url": string
        }
    ]
}

```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the ERRORDOCUMENT section configuration items.

## error\_document\_list

The following is a list of ERRORDOCUMENT settings.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### error\_document\_list/name (Required)

The name of the ERRORDOCUMENT setting. This 'name' must be set when using the functionality of the ERRORDOCUMENT section in other sections.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### error\_document\_list/status (Required)

Specifies the HTTP status code value.

| Item      | Description |
|-----------|-------------|
| Data Type | Integer     |
| Range     | 100 ~ 599   |

### error\_document\_list/url (Required)

Specifies a relative path under the doc\_root or a full path that the client can interpret.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## 3.6.2. Example

The following is an example of configuring the ERRORDOCUMENT section:

```
{
  "error_document": {
    "error_document_list": [
      {
        "name": "forbidden",
        "status": 403,
        "url": "err/403.html"
      }
    ]
  }
}
```

```

    },
    {
      "name": "notfound",
      "status": 404,
      "url": "http://tmaxsoft.co.kr/404.html"
    }
  ]
}

```

## 3.7. FILTER Section

The FILTER section specifies the FILTER section name that will use the Filter module. The Filter item is set in the NODE or SERVER section.

### 3.7.1. Configuration Items

The following is the configuration format of the FILTER section.

```

#"filter": {
  #"filters": [
    {
      "name": string,
      "path": string,          # $ENV
      #"event": [string]
    }
  ]
}

```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the FILTER section configuration items.

#### filters

The following is a list of FILTER settings.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

#### filters/name (Required)

The name of the FILTER setting. This 'name' must be set when using the functionality of the FILTER section in other sections.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### filters/path (Required)

Specifies the path of the Filter module within the physical directory of the server. Environment variables can be used when this setting is configured.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### filters/event

Configures the timing at which the Filter module operates.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

The following describes the filter events by type. Depending on the filter event type, the location is set in either the NODE section or the SERVER section.

- SYSTEM Filter Event

Filters containing SYSTEM filter events are configured in the NODE section.

| Filter Event | Description                                    |
|--------------|------------------------------------------------|
| ON_STOP      | Operates when the WebtoB engine is terminated. |
| ON_START     | Operates when the WebtoB engine starts.        |

- HTTP Filter Events

Filters containing HTTP filter events are configured in the SERVER section.

| Filter Event        | Description                                                                                    |
|---------------------|------------------------------------------------------------------------------------------------|
| RECEIVE_REQUEST     | Operates when an HTTP request is received from the client.                                     |
| BEFORE_SEND_REQUEST | Operates before the HTTP request received from the client is forwarded to the internal server. |
| AFTER_SEND_REQUEST  | Operates after the HTTP request received from the client is forwarded to the internal server.  |

| Filter Event         | Description                                                                                     |
|----------------------|-------------------------------------------------------------------------------------------------|
| RECEIVE_RESPONSE     | Operates when an HTTP response is received from the internal server.                            |
| BEFORE_SEND_RESPONSE | Operates before the HTTP response received from the internal server is forwarded to the client. |
| AFTER_SEND_RESPONSE  | Operates after the HTTP response received from the internal server is forwarded to the client.  |

### 3.7.2. Example

The following is an example configuration for implementing a Filter.

- Filter Project Structure

The following shows the structure of a project to implement a Filter. Each file contains the elements required to define and implement a Filter.

```
.
├── CMakeLists.txt
├── SampleSystemFilter.h
├── SampleSystemFilter.cpp
├── SampleHttpFilter.h
└── SampleHttpFilter.cpp
```

The configuration examples for each file are as follows:

- CMakeLists.txt

```
cmake_minimum_required(VERSION 3.15)

project(sampleFilter)
set(CMAKE_VERBOSE_MAKEFILE true)
set(CMAKE_CXX_STANDARD 17)

message("\t MODULE: ${PROJECT_NAME}")

# Webtob Filter Include Directory
include_directories(/WEBTOB_BINARY_PATH/include)

# Library Add
add_library(sampleSystemFilter SHARED SampleSystemFilter.cpp)
add_library(sampleHttpFilter SHARED SampleHttpFilter.cpp)
```

- SampleSystemFilter.h

```
/**
 * @File      : SampleFilter.h
 * @Author    : Webtob
```

```

* @Year      : 2024
*/

#ifndef WEBTOB_SAMPLE_SYSTEM_FILTER_H
#define WEBTOB_SAMPLE_SYSTEM_FILTER_H

#include <iostream>
#include <string>
#include <sstream>
#include <memory>
#include <vector>
#include "webtob-filter/filter-api/WebtobFilter.h"

namespace webtob {
    class SampleSystemFilter : public WebtobFilter {
    public:
        static std::vector<std::string> calledEvents;

        FilterResult onStart() override;

        FilterResult onStop() override;

        FilterResult onReceiveRequest(const std::shared_ptr<FilterRequest>& request,
                                      const std::shared_ptr<FilterResponse>& response)
        override;

        FilterResult onReceiveResponse(const std::shared_ptr<FilterRequest>& request,
                                       const std::shared_ptr<FilterResponse>& response)
        override;

    private:
        static std::string getStaticVector();

        void printCalledInfo(const char* calledEvent);

        std::string filterName = "SampleSystemFilter";
    };
} // namespace webtob

#endif //WEBTOB6_SAMPLE_SYSTEM_FILTER_H

```

- SampleSystemFilter.cpp

```

#include "SampleSystemFilter.h"

using namespace webtob;

std::vector<std::string> SampleSystemFilter::calledEvents;

// Filter Creator
extern "C" [[maybe_unused]] std::shared_ptr<WebtobFilter> createFilter() {
    return std::make_shared<SampleSystemFilter>();
}

std::string SampleSystemFilter::getStaticVector() {
    std::ostringstream description;

```

```

        description << "Sample SystemFilter Event Called Vector= [\n";
        for (const std::string& event : calledEvents) {
            description << "\t" << event << "\n";
        }
        description << "]\n";
        return description.str();
    }

void SampleSystemFilter::printCalledInfo(const char* calledEvent) {
    calledEvents.emplace_back(calledEvent);
    std::ostringstream message;
    message << "\n=====";
    message << "\n" << filterName << " : " << calledEvent;
    message << "\n-----";
    message << "\n" << getStaticVector();
    message << "\n=====";
    std::cout << message.str() << std::endl;
}

FilterResult SampleSystemFilter::onStart() {
    printCalledInfo(__FUNCTION__);
    return FilterResult::SUCCESS;
}

FilterResult SampleSystemFilter::onStop() {
    printCalledInfo(__FUNCTION__);
    return FilterResult::SUCCESS;
}

FilterResult SampleSystemFilter::onReceiveRequest(const std::shared_ptr<FilterRequest>&
request,
                                                    const std::shared_ptr<FilterResponse>&
response) {
    printCalledInfo(__FUNCTION__);
    return FilterResult::SUCCESS;
}

FilterResult SampleSystemFilter::onReceiveResponse(const std::shared_ptr<FilterRequest>&
request,
                                                    const std::shared_ptr<FilterResponse>&
response) {
    printCalledInfo(__FUNCTION__);
    return FilterResult::SUCCESS;
}

```

#### ◦ SampleHttpFilter.h

```

/**
 * @File      : SampleHttpFilter.h
 * @Author    : Webtob
 * @Year      : 2024
 */
#ifndef WEBTOB6_SAMPLE_HTTP_FILTER_H
#define WEBTOB6_SAMPLE_HTTP_FILTER_H

```

```

#include <iostream>
#include <sstream>
#include "webtob-filter/filter-api/WebtobFilter.h"

namespace webtob {
    class SampleHttpFilter : public WebtobFilter {
    public:
        FilterResult onReceiveRequest(const std::shared_ptr<FilterRequest>& request,
                                     const std::shared_ptr<FilterResponse>& response)
        override;

        FilterResult onBeforeSendRequest(const std::shared_ptr<FilterRequest>& request,
                                         const std::shared_ptr<FilterResponse>& response)
        override;

        FilterResult onAfterSendRequest(const std::shared_ptr<FilterRequest>& request,
                                        const std::shared_ptr<FilterResponse>& response)
        override;

        FilterResult onReceiveResponse(const std::shared_ptr<FilterRequest>& request,
                                       const std::shared_ptr<FilterResponse>& response)
        override;

        FilterResult onBeforeSendResponse(const std::shared_ptr<FilterRequest>& request,
                                          const std::shared_ptr<FilterResponse>& response)
        override;

        FilterResult onAfterSendResponse(const std::shared_ptr<FilterRequest>& request,
                                         const std::shared_ptr<FilterResponse>& response)
        override;

        std::string filterName = "SampleHttpFilter";
    private:
        void printCalledInfo(const char* calledEvent);
    };
} // namespace webtob

#endif //WEBTOB6_SAMPLE_HTTP_FILTER_H

```

#### ◦ SampleHttpFilter.cpp

```

#include "SampleHttpFilter.h"
#include <string_view>

using namespace webtob;

// Filter Creator
extern "C" [[maybe_unused]] std::shared_ptr<WebtobFilter> createFilter() {
    return std::make_shared<SampleHttpFilter>();
}

bool startsWith(const std::string_view& str, const std::string_view& prefix) {
    return str.substr(0, prefix.size()) == prefix;
}

void SampleHttpFilter::printCalledInfo(const char* calledEvent) {

```

```

std::ostringstream message;
message << "\n===== ";
message << "\n" << filterName << " : " << calledEvent;
message << "\n===== ";
std::cout << message.str() << std::endl;
}

FilterResult SampleHttpFilter::onReceiveRequest(const std::shared_ptr<FilterRequest>&
request,
                                                const std::shared_ptr<FilterResponse>&
response) {

    printCalledInfo(__FUNCTION__);

    std::string path = request->getPath();
    if (startsWith(path, "/service/")) {
        if (startsWith(path, "/service/A")) {
            response->setStatusCode(200);
            response->setBody("Filter makes service A response");
        } else if (startsWith(path, "/service/B")) {
            response->setStatusCode(200);
            response->setBody("Service B response");
        } else if (startsWith(path, "/service/C")) {
            response->setStatusCode(400);
            response->setBody("BAD Response Makes");
        }
        return FilterResult::TERMINATE;
    }

    auto queryString = request->getQueryString();

    if (queryString == "test=reject1") {
        response->setStatusCode(200);
        response->setBody("I decide to reject");
        response->addHeader("ErrorCheck", "Reject1_Detected");
        return FilterResult::TERMINATE;
    }
    request->addHeader("Filter", "InsertOn");
    request->addCookie("FilterCookieKey", "OK_Inserted");
    return FilterResult::SUCCESS;
}

FilterResult SampleHttpFilter::onBeforeSendRequest(const std::shared_ptr<FilterRequest>&
request,
                                                const std::shared_ptr<FilterResponse>&
response) {
    printCalledInfo(__FUNCTION__);
    auto queryString = request->getQueryString();
    if (queryString == "test=reject2") {
        response->setStatusCode(200);
        response->setBody("I decide to reject onBeforeSendRequest");
        response->addHeader("ErrorCheck", "onBeforeSendRequest");
        return FilterResult::TERMINATE;
    }

    return FilterResult::SUCCESS;
}

FilterResult SampleHttpFilter::onAfterSendRequest(const std::shared_ptr<FilterRequest>&

```

```

request,
                                const std::shared_ptr<FilterResponse>&
response) {
    printCalledInfo(__FUNCTION__);
    auto queryString = request->getQueryString();
    if (queryString == "test=reject3") {
        response->addHeader("ErrorCheck", "onAfterSendRequest");
        return FilterResult::TERMINATE;
    }
    return FilterResult::SUCCESS;
}

FilterResult SampleHttpFilter::onReceiveResponse(const std::shared_ptr<FilterRequest>&
request,
                                                const std::shared_ptr<FilterResponse>&
response) {
    printCalledInfo(__FUNCTION__);
    auto cookieBuilder = response->getCookieBuilder();
    auto cookie = cookieBuilder-
>newCookie().setName("Cookie_ReceiveResponse").setValue("AddFromResponse")
        .setHttpOnly(true).setPath("/").setSameSite(FilterCookie::SameSite::OFF).build();
    response->addCookie(cookie);
    response->addHeader("Filter_response", "OK_ADDED");
    return FilterResult::SUCCESS;
}

FilterResult SampleHttpFilter::onBeforeSendResponse(const std::shared_ptr<FilterRequest>&
request,
                                                    const std::shared_ptr<FilterResponse>&
response) {
    printCalledInfo(__FUNCTION__);
    auto cookieBuilder = response->getCookieBuilder();
    auto cookie = cookieBuilder->newCookie().setName("MY_COOKIE").setValue("AddFromResponse")
        .setHttpOnly(true).setPath("/").setSameSite(FilterCookie::SameSite::OFF).build();
    response->addCookie(cookie);

    return FilterResult::SUCCESS;
}

FilterResult SampleHttpFilter::onAfterSendResponse(const std::shared_ptr<FilterRequest>&
request,
                                                    const std::shared_ptr<FilterResponse>&
response) {
    printCalledInfo(__FUNCTION__);
    return FilterResult::SUCCESS;
}

```

- Build

```

~/workspace/myTestFilter> cmake . -B build
-- The C compiler identification is AppleClang 15.0.0.15000309
-- The CXX compiler identification is AppleClang 15.0.0.15000309
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working C compiler: /Library/Developer/CommandLineTools/usr/bin/cc - skipped
-- Detecting C compile features
-- Detecting C compile features - done

```

```
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /Library/Developer/CommandLineTools/usr/bin/c++ - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
    MODULE: sampleFilter
-- Configuring done (0.5s)
-- Generating done (0.0s)
-- Build files have been written to: /Users/seungwook/workspace/myTestFilter/build
~/workspace/myTestFilter>

~/workspace/myTestFilter> cd build && make
```

- Configuration

The following is an example of setting the FILTER section in the environment configuration file.

```
{
  "filter": {
    "filters": [
      {
        "name": "filter1",
        "path": "TargetRealPath/myTestFilter/libsampleSystemFilter.so",
        "event": [ "ON_START", "ON_STOP" ]
      },
      {
        "name": "filter2",
        "path": "TargetRealPath/myTestFilter/libsampleHttpFilter.so",
        "event": [ "RECEIVE_REQUEST", "RECEIVE_RESPONSE" ]
      },
    ]
  }
}
```

## 3.8. HEADERS Section

The HEADERS section is used when a specific HTTP Header is changed based on user requests or responses. Headers items can be configured in the SERVER or DESTINATION sections.

### 3.8.1. Configuration Items

The following is the configuration format of the HEADERS section.

```
#"headers": {
  #"headers_list": [
    {
      "name": string,
      "action": string,
      "field_name": string,
      #"field_value": string,
```

```

    # "reg_exp": string,
    # "status_code": integer    # 0
  }
]
}

```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the HEADERS section configuration items.

## headers\_list

The following is a list of HEADERS settings.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### headers\_list/name (Required)

The name of the HEADERS setting. This 'name' must be set when using the functionality of the HEADERS section in other sections.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### headers\_list/action (Required)

Configures the action for handling HTTP headers.

| Item      | Description                                                                                                                                       |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Type | String                                                                                                                                            |
| Range     | "AddRequest"   "AddResponse"   "AddIfAbsentRequest"   "AddIfAbsentResponse"   "AppendResponse"   "EchoResponse"   "SetResponse"   "UnsetResponse" |

The following describes each action type.

| Action     | Description                                                                                                                                    |
|------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| AddRequest | Adds a specified HTTP header in the request. Caution is advised as duplicates may occur if the specified header already exists in the request. |

| Action              | Description                                                                                                                                                      |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| AddResponse         | Adds a specified HTTP header in the response. Caution is advised as duplicates may occur if the specified header already exists in the response.                 |
| AddIfAbsentRequest  | Adds a specified HTTP header when the request contains no HTTP headers.                                                                                          |
| AddIfAbsentResponse | Adds a specified HTTP header when the response contains no HTTP headers.                                                                                         |
| AppendResponse      | Appends a specified FieldValue at the end of an existing header value. If there is no existing header, no action occurs.                                         |
| EchoResponse        | Adds the same header and value to the response as the request, if the request contains a specified header. This action has no FieldValue.                        |
| SetResponse         | Adds a specified HTTP header when the response contains no HTTP headers. If the header already exists, the header value is replaced by the specified FieldValue. |
| UnsetResponse       | Deletes a specified header if it already exists in the response. This action has no FieldValue.                                                                  |



Certain Action types have required field\_value.

### headers\_list/field\_name (Required)

Specifies the name of the HTTP header.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### headers\_list/field\_value

Specifies the value of the HTTP header.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### headers\_list/reg\_exp

Set to control headers when the HTTP request URL matches a Regular expression pattern.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 511 characters |

### headers\_list/status\_code

Set to control headers for a specific HTTP status code.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 599     |
| Default Value | 0           |

### 3.8.2. Example

The following is an example of configuring the HEADERS section:

```
{
  "headers": {
    "headers_list": [
      {
        "name": "header1",
        "action": "AppendResponse",
        "field_name": "Test_Header",
        "field_value": "test"
      }
    ]
  }
}
```

## 3.9. LOGGING Section

The LOGGING section configures client request log. Access log and error log are recorded separately, and their format can be configured. System logs, access logs, and error logs are all set in the LOGGING section.

### 3.9.1. Configuration Items

The following is the configuration format of the LOGGING section.

```
"logging": {
  "system_log": [
    {
```

```

    "name": string,
    #"level": string,                                # "INFO"
    #"dump": [string],
    #"handlers": {
        #"file_handler": {
            "file_name": string,                    # "webtob_system.log", $ENV
            #"rotate_by_seconds": integer,           # 0
            #"valid_hours": integer,                 # 0
            #"rotate_by_file_size": integer,         # 0
            #"archive_file_name": string,            # $ENV
            #"enable_sync": boolean,                  # false
            #"permission": string,                   # "0600"
            #"rotation_max_file_count": integer      # 0
        },
        #"enable_console_handler": boolean          # false
    }
}
],
#"access_log": [
{
    "name": string,
    #"level": string,                                # "INFO"
    #"format": string,                               # "DEFAULT"
    #"exclude_by_ext": string,
    #"handlers": {
        #"file_handler": {
            "file_name": string,                    # "webtob_access.log", $ENV
            #"rotate_by_seconds": integer,           # 0
            #"valid_hours": integer,                 # 0
            #"rotate_by_file_size": integer,         # 0
            #"archive_file_name": string,            # $ENV
            #"enable_sync": boolean,                  # false
            #"permission": string,                   # "0600"
            #"rotation_max_file_count": integer      # 0
        },
        #"enable_console_handler": boolean          # false
    }
}
],
#"error_log": {
    #"level": "string",                              # "INFO"
    #"format": "string",                             # "ERROR"
    #"enable_exclude_client_address_on_error": boolean, # false
    #"handlers": {
        #"file_handler": {
            "file_name": string,                    # "webtob_error.log", $ENV
            #"rotate_by_seconds": integer,           # 0
            #"valid_hours": integer,                 # 0
            #"rotate_by_file_size": integer,         # 0
            #"archive_file_name": string,            # $ENV
            #"enable_sync": boolean,                  # false
            #"permission": string,                   # "0600"
            #"rotation_max_file_count": integer      # 0
        },
        #"enable_console_handler": boolean          # false
    }
}
},
}

```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the LOGGING section configuration items.

## system\_log (Required)

Configures the system log.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

## system\_log/name (Required)

Specifies the target system logger to be set.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

Logger names are hierarchical and separated by a period (.). The settings for loggers prioritize the lowest-level configuration. For example, if both (A)'webtob' and (B)'webtob.http' are configured, the 'webtob.http' logger follows the configuration (B), while other loggers follow the configuration (A). The top-level system logger 'webtob' must be configured.

The following describes the loggers that can be configured.

| Logger Name        | Description                                                          |
|--------------------|----------------------------------------------------------------------|
| webtob.http        | Manages logs related to handling HTTP messages.                      |
| webtob.wjp         | Manages logs related to handling WJP messages.                       |
| webtob.ssl         | Manages logs related to SSL.                                         |
| webtob.network     | Manages logs related to network operations.                          |
| webtob.server      | Manages logs related to the startup and termination of each thread.  |
| webtob.em.acceptor | Manages logs related to the Acceptor, which handles new connections. |
| webtob.em.control  | Manages logs related to Admin API communication.                     |
| webtob.em.worker   | Manages logs related to HTH.                                         |
| webtob.util        | Manages logs related to other utility functions.                     |

The following is a configuration example:

```
"system_log": [  
  {  
    "name": "webtob",  
    "level": "DEBUG",
```

```

    "handlers": {
      "file_handler": {
        "file_name": "webtob.log"
      },
      "enable_console_handler": false
    },
    "dump": [
      "ToClient",
      "FromClient",
      "ToServer",
      "FromServer",
      "SSLRead",
      "SSLWrite"
    ]
  },
  {
    "name": "webtob.http",
    "level": "DEBUG",
    "handlers": {
      "file_handler": {
        "file_name": "webtob_http.log"
      },
      "enable_console_handler": false
    }
  }
]

```

## system\_log/level

Specifies the log level.

If the level of the system log message is higher than the set value, it will be output. The supported log levels from lowest to highest are "DEBUG", "INFO", "WARNING", "FATAL", and "OFF".

| Item          | Description                                    |
|---------------|------------------------------------------------|
| Data Type     | String                                         |
| Range         | "DEBUG"   "INFO"   "WARNING"   "FATAL"   "OFF" |
| Default Value | "INFO"                                         |



The lower the level, the more detailed the log messages are output.

## system\_log/dump

Configures whether to output dump logs. This setting applies only when the log level is DEBUG, and can only be set for loggers named "webtob".

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (string) |
| Range     | Up to 6 items  |

The following describes each configuration value.

| Value      | Description                                         |
|------------|-----------------------------------------------------|
| FromClient | Logs data sent by the client.                       |
| ToClient   | Logs data sent to the client.                       |
| FromServer | Logs data sent by the server.                       |
| ToServer   | Logs data sent to the server.                       |
| SSLRead    | Logs data read with SSL encryption.                 |
| SSLWrite   | Logs data that will be written with SSL encryption. |

## system\_log/handlers

Sets the file and console to which log messages will be recorded.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## system\_log/handlers/file\_handler

Sets the file to which logs will be recorded.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## system\_log/handlers/file\_handler/file\_name (Required)

Sets the path to the file where the system log will be stored. Environment variables can be used when this setting is configured.

If the relative path does not start with a slash (/), it is automatically replaced with "\$WEBTOB\_HOME/relative path".

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "webtob_system.log"  |

The following substitution strings specified in the file name are replaced with the actual values when the file is created. For example, a value set to "log/system\_%Y%%M%%D%.log" in FileName can be replaced with "\$WEBTOB\_HOME/log/system\_20231023.log" when the log file is created.

The following describes each substitution string:

| String     | Actual Value                                                         |
|------------|----------------------------------------------------------------------|
| %Y%        | 4-digit year (Example: 2009)                                         |
| %M%        | 2-digit month (Example: 11)                                          |
| %D%        | 2-digit day (Example: 05)                                            |
| %h%        | 2-digit hour (Example: 10)                                           |
| %m%        | 2-digit minute (Example: 30)                                         |
| %s%        | 2-digit second (Example: 45)                                         |
| %pid%      | The current process ID. (Example: 12345)                             |
| %name%     | Name of the current process. (Example: Engine)                       |
| %fullname% | Name of the current process including the index. (Example: Engine-0) |

### system\_log/handlers/file\_handler/rotate\_by\_seconds

Specifies the maximum period of time to create a new log file. If an existing log file is older than this period of time, a new log file is created to record a new log.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 86400   |
| Default Value | 0           |

### system\_log/handlers/file\_handler/valid\_hours

Specifies the creation of a new log file at the specified interval.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 23      |
| Default Value | 0           |

### system\_log/handlers/file\_handler/rotate\_by\_file\_size

Specifies the file size threshold for log messages. If an existing log file surpasses this specified size, a new log file will be created.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **system\_log/handlers/file\_handler/archive\_file\_name**

Specifies the log file name format for an archived file. When a new log file is created, the existing log file is renamed as set in archive\_file\_name. New log files will then be created in the format specified in file\_name. Environment variables can be used when this setting is configured.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **system\_log/handlers/file\_handler/enable\_sync**

Specifies whether to write to the file immediately when logs are recorded.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following describes each configuration value.

| Value | Description                                                                                                                                                                                           |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| True  | Log messages are written directly to files without being buffered in WebtoB memory.<br><br><b>[Note]</b> Services that require immediate logs should use this feature to immediately identify errors. |

### **system\_log/handlers/file\_handler/permission**

Specifies the permission for system log files.

This configuration is the same as the file access permissions used in UNIX and can be used only in a UNIX/Linux environment.

| Item      | Description |
|-----------|-------------|
| Data Type | String      |

| Item  | Description     |
|-------|-----------------|
| Range | "0600" ~ "0777" |

### system\_log/handlers/file\_handler/rotation\_max\_file\_count

Sets the maximum number of log files to create when using the system log file rotation feature. If set to 0, there is no limit on the number of log files generated by the rotation feature.



When the number of log files exceeds the specified value, the oldest log files are deleted first.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 9999    |
| Default Value | 0           |

### system\_log/handlers/enable\_console\_handler

Specifies whether to display the log in the console.

| Item      | Description |
|-----------|-------------|
| Data Type | Boolean     |
| Range     | False       |

### access\_log (Required)

Configures access logs.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### access\_log/name

Sets the name of the access logger. This name is used to specify the target for logging access logs in other sections, such as the HTTP section.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## access\_log/level

Sets the access log level.

| Item          | Description              |
|---------------|--------------------------|
| Data Type     | String                   |
| Range         | "TRACE"   "INFO"   "OFF" |
| Default Value | "INFO"                   |

The following describes each configuration value.

| Value | Description                                              |
|-------|----------------------------------------------------------|
| OFF   | Does not output access logs.                             |
| INFO  | Outputs basic access logs.                               |
| TRACE | Outputs additional logs for the request processing flow. |

## access\_log/format

Sets the format of the messages to be recorded in the access log file.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

The following describes each configuration value.

| Value      | Description                                                                                                      |
|------------|------------------------------------------------------------------------------------------------------------------|
| DEFAULT    | Default log file format.<br>(Log format: "%h %t \"%r\" %s %b %D")                                                |
| COMMON     | Common log file format.<br>(Log format: "%h %l %u %t \"%r\" %s %b")                                              |
| COMBINED   | Combined log file format.<br>(Log format: "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\"")         |
| COMBINEDIO | CombinedIO log file format.<br>(Log format: "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O") |
| ERROR      | Error log file format.<br>(Log format: "%r")                                                                     |
| %a         | Displays the IP address of the device that sent the request. Same as %h.                                         |
| %A         | Displays the IP address of the server.                                                                           |
| %b         | Displays the bytes of the response excluding headers.                                                            |

| Value            | Description                                                                                           |
|------------------|-------------------------------------------------------------------------------------------------------|
| %{_attr_name_}C  | Displays the value corresponding to '_attr_name_' among the cookie header values of the HTTP request. |
| %d               | Displays the time the response was sent.                                                              |
| %D               | Displays the time taken to process the request (unit: milliseconds).                                  |
| %{ENV_NAME}e     | Displays the environment variable ENV_NAME.                                                           |
| %h               | Displays the IP address of the device that sent the request. Same as %a.                              |
| %H               | Displays the HTTP version used.                                                                       |
| %{HEADER_FIELD}i | Displays the value of the HEADER_FIELD header of an HTTP request.                                     |
| %I               | Displays the bytes of the request.                                                                    |
| %l               | Displays the name for remote login.                                                                   |
| %m               | Displays the HTTP request method.                                                                     |
| %O               | Displays the bytes of the response.                                                                   |
| %p               | Displays the server port number that received the request.                                            |
| %q               | Displays the query value of the HTTP request.                                                         |
| %r               | Displays the entire request line of the HTTP request.                                                 |
| %s               | Displays the HTTP status code used in the response.                                                   |
| %S               | Displays HTTP and HTTPS.                                                                              |
| %t               | Displays the time when the request processing is completed.                                           |
| %T               | Displays the time taken to process the request (unit: seconds).                                       |
| %u               | Displays the username used for HTTP authentication.                                                   |
| %U               | Displays the HTTP request URI.                                                                        |
| %v               | Displays the Host Header field value.                                                                 |

### access\_log/exclude\_by\_ext

Specifies the file extensions to exclude from access logging.

| Value     | Description           |
|-----------|-----------------------|
| Data Type | string                |
| Range     | Up to 1023 characters |

### access\_log/handlers

Sets the file and console to which log messages will be recorded.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## **access\_log/handlers/file\_handler**

Sets the file to which logs will be recorded.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## **access\_log/handlers/file\_handler/file\_name (Required)**

Sets the path to the file where the access log will be stored. Environment variables can be used when this setting is configured.

If the relative path does not start with a slash (/), it is automatically replaced with "\$WEBTOB\_HOME/relative path".

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "webtob_access.log"  |

The following substitution strings specified in the file name are replaced with the actual values when the file is created. For example, a value set to "log/access\_%Y%%M%%D%.log" in FileName can be replaced with "\$WEBTOB\_HOME/log/access\_20231023.log" when the log file is created.

The following describes each substitution string:

| String | Actual Value                             |
|--------|------------------------------------------|
| %Y%    | 4-digit year (Example: 2009)             |
| %M%    | 2-digit month (Example: 11)              |
| %D%    | 2-digit day (Example: 05)                |
| %h%    | 2-digit hour (Example: 10)               |
| %m%    | 2-digit minute (Example: 30)             |
| %s%    | 2-digit second (Example: 45)             |
| %pid%  | The current process ID. (Example: 12345) |

## **access\_log/handlers/file\_handler/rotate\_by\_seconds**

Specifies the maximum period of time to create a new log file. If an existing log file is older than this

period of time, a new log file is created to record a new log.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 86400   |
| Default Value | 0           |

### **access\_log/handlers/file\_handler/valid\_hours**

Specifies the creation of a new log file at the specified interval.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 23      |
| Default Value | 0           |

### **access\_log/handlers/file\_handler/rotate\_by\_file\_size**

Specifies the file size threshold for log messages. If an existing log file surpasses this specified size, a new log file will be created.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### **access\_log/handlers/file\_handler/archive\_file\_name**

Specifies the log file name format for an archived file. When a new log file is created, the existing log file is renamed as set in archive\_file\_name. New log files will then be created in the format specified in file\_name. Environment variables can be used when this setting is configured.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **access\_log/handlers/file\_handler/enable\_sync**

Specifies whether to write to the file immediately when logs are recorded.

| Item      | Description |
|-----------|-------------|
| Data Type | Boolean     |
| Range     | False       |

The following describes each configuration value.

| Value | Description                                                                                                                                                                                           |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| True  | Log messages are written directly to files without being buffered in WebtoB memory.<br><br><b>[Note]</b> Services that require immediate logs should use this feature to immediately identify errors. |

### access\_log/handlers/file\_handler/permission

Specifies the permission for access log files.

This configuration is the same as the file access permissions used in UNIX and can be used only in a UNIX/Linux environment.

| Item      | Description     |
|-----------|-----------------|
| Data Type | String          |
| Range     | "0600" ~ "0777" |

### access\_log/handlers/file\_handler/rotation\_max\_file\_count

Sets the maximum number of log files to create when using the access log file rotation feature. If set to 0, there is no limit on the number of log files generated by the rotation feature.



When the number of log files exceeds the specified value, the oldest log files are deleted first.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 9999    |
| Default Value | 0           |

### access\_log/handlers/enable\_console\_handler

Specifies whether to display the log in the console.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **error\_log (Required)**

Sets up error logs.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |

### **error\_log/level**

Sets the error log level.

| Item          | Description    |
|---------------|----------------|
| Data Type     | String         |
| Range         | "INFO"   "OFF" |
| Default Value | "INFO"         |

The following describes each configuration value.

| Value | Description                 |
|-------|-----------------------------|
| OFF   | Does not output error logs. |

### **error\_log/format**

Sets the format of the messages to be recorded in the error log file.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

The following describes each configuration value.

| Value   | Description                                                         |
|---------|---------------------------------------------------------------------|
| DEFAULT | Default log file format.<br>(Log format: "%h %t \"%r\" %s %b %D")   |
| COMMON  | Common log file format.<br>(Log format: "%h %l %u %t \"%r\" %s %b") |

| Value            | Description                                                                                                      |
|------------------|------------------------------------------------------------------------------------------------------------------|
| COMBINED         | Combined log file format.<br>(Log format: "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\"")         |
| COMBINEDIO       | CombinedIO log file format.<br>(Log format: "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O") |
| ERROR            | Error log file format.<br>(Log format: "%r")                                                                     |
| %a               | Displays the IP address of the device that sent the request. Same as %h.                                         |
| %A               | Displays the IP address of the server.                                                                           |
| %b               | Displays the bytes of the response excluding headers.                                                            |
| %{_attr_name_}C  | Displays the value corresponding to '_attr_name_' among the cookie header values of the HTTP request.            |
| %d               | Displays the time the response was sent.                                                                         |
| %D               | Displays the time taken to process the request (unit: milliseconds).                                             |
| %{ENV_NAME}e     | Displays the environment variable ENV_NAME.                                                                      |
| %h               | Displays the IP address of the device that sent the request. Same as %a.                                         |
| %H               | Displays the HTTP version used.                                                                                  |
| %{HEADER_FIELD}i | Displays the value of the HEADER_FIELD header of an HTTP request.                                                |
| %I               | Displays the bytes of the request.                                                                               |
| %l               | Displays the name for remote login.                                                                              |
| %m               | Displays the HTTP request method.                                                                                |
| %O               | Displays the bytes of the response.                                                                              |
| %p               | Displays the server port number that received the request.                                                       |
| %q               | Displays the query value of the HTTP request.                                                                    |
| %r               | Displays the entire request line of the HTTP request.                                                            |
| %s               | Displays the HTTP status code used in the response.                                                              |
| %S               | Displays HTTP and HTTPS.                                                                                         |
| %t               | Displays the time when the request processing is completed.                                                      |
| %T               | Displays the time taken to process the request (unit: seconds).                                                  |
| %u               | Displays the username used for HTTP authentication.                                                              |
| %U               | Displays the HTTP request URI.                                                                                   |
| %v               | Displays the host header field value.                                                                            |

## error\_log/enable\_exclude\_client\_address\_on\_error

Excludes the client IP address from error messages logged by WebtoB. This can prevent the exposure of user IP information, enhancing security.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

## error\_log/handlers

Sets the file and console to which log messages will be recorded.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## error\_log/handlers/file\_handler

Sets the file to which logs will be recorded.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## error\_log/handlers/file\_handler/file\_name (Required)

Sets the path to the file where the error log will be stored. Environment variables can be used when this setting is configured.

If the relative path does not start with a slash (/), it is automatically replaced with "\$WEBTOB\_HOME/relative path".

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "webtob_error.log"   |

The following substitution strings specified in the file name are replaced with the actual values when the file is created. For example, a value set to "log/error\_%Y%%M%%D%.log" in FileName can be replaced with "\$WEBTOB\_HOME/log/error\_20231023.log" when the log file is created.

The following describes each substitution string:

| String | Actual Value                 |
|--------|------------------------------|
| %Y%    | 4-digit year (Example: 2009) |

| String | Actual Value                             |
|--------|------------------------------------------|
| %M%    | 2-digit month (Example: 11)              |
| %D%    | 2-digit day (Example: 05)                |
| %h%    | 2-digit hour (Example: 10)               |
| %m%    | 2-digit minute (Example: 30)             |
| %s%    | 2-digit second (Example: 45)             |
| %pid%  | The current process ID. (Example: 12345) |

### **error\_log/handlers/file\_handler/rotate\_by\_seconds**

Specifies the maximum period of time to create a new log file. If an existing log file is older than this period of time, a new log file is created to record a new log.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 86400   |
| Default Value | 0           |

### **error\_log/handlers/file\_handler/valid\_hours**

Specifies the creation of a new log file at the specified interval.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 23      |
| Default Value | 0           |

### **error\_log/handlers/file\_handler/rotate\_by\_file\_size**

Specifies the file size threshold for log messages. If an existing log file surpasses this specified size, a new log file will be created.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

### error\_log/handlers/file\_handler/archive\_file\_name

Specifies the log file name format for an archived file. When a new log file is created, the existing log file is renamed as set in archive\_file\_name. New log files will then be created in the format specified in file\_name. Environment variables can be used when this setting is configured.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### error\_log/handlers/file\_handler/enable\_sync

Specifies whether to write to the file immediately when logs are recorded.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following describes each configuration value.

| Value | Description                                                                                                                                                                                           |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| True  | Log messages are written directly to files without being buffered in WebtoB memory.<br><br><b>[Note]</b> Services that require immediate logs should use this feature to immediately identify errors. |

### error\_log/handlers/file\_handler/permission

Specifies the permission for error log files.

This configuration is the same as the file access permissions used in UNIX and can be used only in a UNIX/Linux environment.

| Item      | Description     |
|-----------|-----------------|
| Data Type | String          |
| Range     | "0600" ~ "0777" |

### error\_log/handlers/file\_handler/rotation\_max\_file\_count

Sets the maximum number of log files to create when using the error log file rotation feature. If set to 0, there is no limit on the number of log files generated by the rotation feature.



When the number of log files exceeds the specified value, the oldest log files are deleted first.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 9999    |
| Default Value | 0           |

### error\_log/handlers/enable\_console\_handler

Specifies whether to display the log in the console.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

## 3.9.2. Example

The following is an example of configuring the LOGGING section:

```
"logging": {
  "system_log": [
    {
      "name": "webtob",
      "level": "DEBUG",
      "handlers": {
        "file_handler": {
          "file_name": "webtob-%Y%M%D%-h%m%$$.log"
        }
      }
    }
  ],
  "access_log": [
    {
      "name": "access_log1",
      "level": "INFO",
      "format": "DEFAULT",
      "handlers": {
        "file_handler": {
          "file_name": "access-%Y%M%D%$.log"
        }
      }
    },
    {
      "name": "access_log2",
      "level": "INFO",
      "format": "DEFAULT",
      "handlers": {
```

```

        "file_handler": {
            "file_name": "access2-%Y%M%D%%s%.log"
        }
    },
    "error_log": {
        "level": "INFO",
        "format": "ERROR",
        "handlers": {
            "file_handler": {
                "file_name": "logs/errors%%s%.log"
            }
        }
    }
}

```

## 3.10. NODE Section

The NODE section configures each node that composes WebtoB. This section affects all operations related to the node.

The NODE section can define 'WebtoB system path' and 'other system configuration'.

### 3.10.1. Configuration Items

The following is the configuration format of the NODE section.

```

"node": {
    "name": string                # $ENV
    # "process_count": integer,    # 1 (1-255)
    # "manager_port": integer,     # 9000 (1-65535)
    # "hth_count": integer,        # 1 (1-255)
    # "user": string,
    # "worker_threads": integer,   # 8 (1-100)
    # "hth_schedule": string,      # "RR"
    # "connection_pool_size": integer, # 8192 (1-INT_MAX)
    # "graceful_shutdown_timeout": integer # 60 (1-INT_MAX)
    # "cache_key": string,         # "HOST_URI"
    # "cache_entry": integer       # 128 (0-INT_MAX)
    # "max_cache_memory_size": integer # 100 (0-INT_MAX)
    # "cache_max_file_size": integer # 8192 (0-INT_MAX)
    # "listen_backlog": integer    # 4096 (0-INT_MAX)
    # "limit_request_body_size": integer # 0 (0-)
    # "limit_request_header_field_count": integer # 100 (0-INT_MAX)
    # "limit_request_header_field_size": integer # 8190 (0-INT_MAX)
    # "limit_request_line_size": integer # 8190 (0-INT_MAX)
    # "system_filters": [string]
}

```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the NODE section configuration items.

## name (Required)

Specifies the node name.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 127 characters |

The following is an example of how to configure the name. If the prompt displays 'user@webtob\_node:~\$', set the name as follows.

Environment variables can be used when configuring this setting. Using \${NODENAME} applies the output of the "hostname" command.

```
"node": {  
  "name": "webtob_node",  
  ...  
}
```



The node name must be the same as the host name. If it is different, it cannot be run.

## process\_count

Specifies the number of processes to handle requests.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ 255     |
| Default Value | 1           |

## manager\_port

Sets the Admin port of the manager that controls the engine.

| Item      | Description |
|-----------|-------------|
| Data Type | Integer     |

| Item          | Description |
|---------------|-------------|
| Range         | 1 ~ 65535   |
| Default Value | 9000        |

## user

Specifies the user account under which WebtoB operates.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |



This setting is applicable only in UNIX/Linux environments. For system security, it is recommended to use a regular user account rather than a root account. For more information, refer to [Engine Privilege Separation](#).

## hth\_count

Specifies the number of HTTP handlers (HTH).

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ 100     |
| Default Value | 1           |

The following is an example of the hth\_count configuration. If 'hth\_count' is set to 3 as follows, 'HTH-0', 'HTH-1', and 'HTH-2' will be created.

```
"node": {
  "name": "webtob_node",
  "hth_count": 3,
  ...
}
```



You can check the generated HTH through the log (webtob.log).

```
EventManager(HTH-2) Thread Created. Thread ID: 4800
```

## worker\_threads

Sets the number of worker threads to assign to HTH. Worker threads process static files, SSL/TLS (handshake, encryption, and decryption), compression, and HTTP authentication.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ 100     |
| Default Value | 8           |

The following is an example of the worker\_threads configuration. If 'worker\_threads' is set to 5 as follows, 'HTMLS-0', 'HTMLS-1', 'HTMLS-2', 'HTMLS-3', and 'HTMLS-4' will be created.

```
"node": {  
  "name": "webtob_node",  
  "hth_count": 3,  
  "worker_threads": 5,  
  ...  
}
```



You can check the created worker thread through the log (webtob.log).

```
EventManager(HTMLS-4) Thread Created. Thread ID: 19271
```



Since the number of FDs used equals  $\text{hth\_count} \times \text{worker\_threads}$ , you must consider the number of FDs when configuring hth\_count and worker\_threads. If the set value exceeds the number of FDs, the server will shut down and an error log occurs as follows:

```
Error: eventFd() failed: Too many open files  
Too many open files
```

## hth\_schedule

Specifies the method to specify the HTH to process the request when there are multiple HTHs.

| Item          | Description |
|---------------|-------------|
| Data Type     | String      |
| Range         | "RR"        |
| Default Value | "RR"        |

The following describes each configuration value.

| Value | Description                                                         |
|-------|---------------------------------------------------------------------|
| RR    | Assigns requests to HTHs sequentially using the round robin method. |

The following is an example of how to configure hth\_schedule:

```
"node": {
  "name": "webtob_node",
  "hth_count": 2,
  "worker_threads": 8,
  "hth_schedule": "RR",
  ...
}
```



You can check in the webtob.log file that requests are assigned sequentially based on the number set in hth\_count.

```
[2024-07-25 06:49:19.113263413][DEBUG][HTH-0][SVR_3455]WJP2RequestMessageHandler.cpp:87 serviceMessage: WJP Request Loop Done.
[2024-07-25 06:49:19.113268083][DEBUG][HTH-0][NET_2004]EventManager.cpp:129 loop: End of this loop.
[2024-07-25 06:49:19.425343033][DEBUG][HTH-1][NET_4000]EpollMultiplexer.cpp:33 select: Multiplexer awakened. Event Count=0.
[2024-07-25 06:49:19.425360647][DEBUG][HTH-1][NET_2004]EventManager.cpp:129 loop: End of this loop.
[2024-07-25 06:49:20.114399067][DEBUG][HTH-0][NET_4000]EpollMultiplexer.cpp:33 select: Multiplexer awakened. Event Count=0.
[2024-07-25 06:49:20.114416261][DEBUG][HTH-0][NET_2004]EventManager.cpp:129 loop: End of this loop.
[2024-07-25 06:49:20.426438758][DEBUG][HTH-1][NET_4000]EpollMultiplexer.cpp:33 select: Multiplexer awakened. Event Count=0.
[2024-07-25 06:49:20.426456148][DEBUG][HTH-1][NET_2004]EventManager.cpp:129 loop: End of this loop.
```

connection\_pool\_size

Sets the size of the pool that manages connections for each HTH.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 8192        |



Similar to hth\_count and worker\_threads, this setting is affected by the number of

FDs, so the number of FDs must be considered. If the set value exceeds the number of FDs, the following error will occur and the server will shut down.

```
Error: eventFd() failed: Too many open files
Too many open files
```

## graceful\_shutdown\_timeout

Sets a timeout to wait before forcing a shutdown after a graceful shutdown command.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ INT_MAX |
| Default Value | 60          |

The following is an example of how to configure graceful\_shutdown\_timeout:

```
{
  "node": {
    "name": "webtob_node",
    "hth_count": 1,
    "worker_threads": 8,
    "graceful_shutdown_timeout": 5
  },
  ...
}
```

## cache\_key

Sets the format of the key used to identify cached data.

| Item          | Description           |
|---------------|-----------------------|
| Data Type     | String                |
| Range         | "HOSTS"   "REAL_PATH" |
| Default Value | "HOST_URI"            |

The following describes each configuration value.

| Value     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| HOST_URI  | <p>Use the request URI and the HOST value of the request header to create a Key.</p> <p>Even if REQUEST_URI is the same, the VHOST depends on HTTP_HOST. This means that the actual path can be different. Although VHOST is different, the same actual path can cause duplicate caching.</p> <p>The response is faster than using REAL_PATH because the real path does not need to be calculated. Therefore, if DocRoot differs by VHOST, it is better to use the HOST_URI.</p> <p>Since responses handled by JEUS or Reverse Proxy cannot be calculated, HOST_URI value is used.</p> |
| REAL_PATH | <p>Uses the actual path of the request URI to generate a Key. This value is applied only when SVRTYPE is HTML and helps to address potential issues that may occur when using HOST_URI.</p> <p>Even if the HTTP_HOST is different, the actual path of the same requested file may be the same. In this case, only one file is cached regardless of the HTTP_HOST, which can reduce memory usage. However, since the actual path needs to be calculated, the response time may be slightly slower than using HOST_URI.</p>                                                              |

The following is an example of how to configure cache\_key. To store the cache, 'enable\_cache' must be set to true in the destination section.

```
"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads": 8,
  "cache_key": "HOST_URI",
  "cache_entry": 128,
  "max_cache_memory_size": 100,
  "cache_max_file_size": 8192
  ...
  "destination": {
    "htmls": [
      {
        "name": "htmls1",
        "enable_cache": true
      }
    ]
  }
  ...
}
```



The cache status can be checked in the wsadmin's cache-list. If the cache\_key setting value is 'HOST\_URI', it will be displayed as 192.168.0.1:80/image.jpg, and if it is 'REAL\_PATH', it will be displayed as /home/tmax/webtob6/docs/image.jpg.

```
[wsadmin]>> cache-list
```

```

| HTH-0: Cache List Info |
|-----|
|           Cache key           | Expired time | Cache size |
|-----|
| 192.168.0.1:80/image.jpg | 2024-01-01 00:00:00 |      300 |
|-----|
| Cache count : 1 |
| Memory usages : 300 |
|-----|

```

## cache\_entry

Sets the size of the hash table Key for the HTH cache.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 128         |

The following describes each configuration value.

| Value | Description                  |
|-------|------------------------------|
| 0     | Does not cache the response. |

The following is an example of how to configure cache\_entry. To store the cache, 'enable\_cache' must be set to true in the destination section.

```

"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads": 8,
  "cache_key": "HOST_URI",
  "cache_entry": 128,
  "max_cache_memory_size": 100,
  "cache_max_file_size": 8192
  ...
  "destination": {
    "htmls": [
      {
        "name": "htmls1",
        "enable_cache": true
      }
    ]
  }
  ...

```



The cache status can be checked in the wsadmin's cache-list. If the cache\_entry setting value is 2, a maximum of 2 rows will be displayed.

```
[wsadmin]>> cache-list
-----
-
| HTH-0: Cache List Info
|
-----
-
|           Cache key           |   Expired time   | Cache size
|
-----
-
| 192.168.0.1:80/image.jpg | 2024-01-01 00:00:24 |    345|
| 192.168.0.1:80/image1.jpg | 2024-01-01 00:00:21 |    345|
|
-----
-
| Cache count : 2
|
| Memory usages : 1035
|
-----
-

```

## max\_cache\_memory\_size

Specifies the maximum amount of memory used by a HTH process for caching.

This is not the total memory size used by all HTH processes, but it applies to each HTH process.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | Mbytes      |
| Range         | 0 ~ INT_MAX |
| Default Value | 100         |

The following describes each configuration value.

| Value | Description                  |
|-------|------------------------------|
| 0     | Does not cache the response. |

The following is an example of how to configure max\_cache\_memory\_size. To store the cache, 'enable\_cache' must be set to true in the destination section.

```
"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads": 8,
  "cache_key": "HOST_URI",
  "cache_entry": 128,

```

```
"max_cache_memory_size":100,
"cache_max_file_size":8192
...
"destination": {
  "htmls": [
    {
      "name": "htmls1",
      "enable_cache":true
    }
  ]
}
...
```

The cache status can be checked in the wsadmin's cache-list. If the max\_cache\_memory\_size setting is 100, the size of images stored in the cache cannot exceed 100 MB.



```
[wsadmin]>> cache-list
-----
| HTH-0: Cache List Info
|
-----
|                               | Expired time   | Cache size
|                               |               |
-----
| 192.168.0.1:80/image.jpg | 2024-01-01 00:00:24 | 345|
| 192.168.0.1:80/image1.jpg | 2024-01-01 00:00:21 | 345|
-----
|
| Cache count : 2
|
| Memory usages : 1035
|
-----
|
|
```

## cache\_max\_file\_size

Specifies the maximum size of a response (headers + body) that can be cached.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ INT_MAX |
| Default Value | 8192        |

The following describes each configuration value.

| Value | Description                  |
|-------|------------------------------|
| 0     | Does not cache the response. |

The following is an example of how to configure `cache_max_file_size`. To store the cache, 'enable\_cache' must be set to true in the destination section.

```
"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads":8,
  "cache_key":"HOST_URI",
  "cache_entry":128,
  "max_cache_memory_size":100,
  "cache_max_file_size":8192
  ...
  "destination": {
    "htmls": [
      {
        "name": "htmls1",
        "enable_cache":true
      }
    ]
  }
  ...
}
```

The cache status can be checked in the wsadmin's cache-list. If the `cache_max_file_size` setting is 8192, images exceeding 8192 bytes will not be cached.



```
[wsadmin]>> cache-list
-----
| HTH-0: Cache List Info
|
-----
|
| Cache key | Expired time | Cache size
|
-----
| 192.168.0.1:80/image.jpg | 2024-01-01 00:00:24 | 345|
| 192.168.0.1:80/image1.jpg | 2024-01-01 00:00:21 | 345|
|
-----
|
| Cache count : 2
|
| Memory usages : 1035
|
-----
|
|
```

## listen\_backlog

Specifies the maximum number of queued pending connections. This limits the number of sockets attempting to access the service port.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 4096        |



This may be related to the number of sockets in SYN\_RECV state in the netstat command.

The following is an example of the listening\_backlog setting.

```
"node": {  
  "name": "webtob_node",  
  "hth_count": 1,  
  "worker_threads": 8,  
  "listen_backlog": 50  
  ...  
}
```



You can check the changed Send-Q using the ss command.

If the listen\_backlog setting is set to 511 or higher, the Send-Q value does not exceed 511. This can be adjusted by changing the system value using the 'sysctl -w net.core.somaxconn=[setting value]'.

```
user@webtob_node:~/tmax/webtob6/$ ss -lnt | grep 80  
State Recv-Q Send-Q Local Address:Port Peer Address:Port  
LISTEN 0 50 0.0.0.0:80 0.0.0.0:*
```

## limit\_request\_body\_size

Specifies the client request body size supported by the server through the HTTP protocol.



If the HTTP request body is 2G (Long type) or larger, it can be processed starting from JEUS version 7 and later.

This value is determined based on the 'Content-Length' value of the HTTP request header, and specifies the maximum body size to allow for chunked requests. If a request body is larger than the set value, the server responds with '413 Request Entity Too Large'.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 or more   |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                        |
|-------|------------------------------------|
| 0     | No limit on the request body size. |

The following is an example of how to configure `limit_request_body_size`.

```
"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads": 8,
  "limit_request_body_size": 100
  ...
}
```

If `limit_request_body_size` is set to 100, the following message is displayed when Content-Length exceeds 100.

```
* Closing connection 0
The body is too large.
```

## limit\_request\_header\_field\_count

Specifies the maximum number of a client request's HTTP header fields.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 100         |

The following describes each configuration value.

| Value | Description                              |
|-------|------------------------------------------|
| 0     | No limit on the number of header fields. |

The following is an example of how to configure `limit_request_header_field_count`.

```
"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads": 8,
  "limit_request_header_field_count": 1
  ...
}
```

When `limit_request_header_field_count` is set to 1, the following message is displayed if the number of header fields exceeds 1.

```
* Closing connection 0
The number of request header fields exceeds the server limit.
```

## limit\_request\_header\_field\_size

Specifies the maximum size of each HTTP header field in a client request.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ 16382   |
| Default Value | 8190        |

The following describes each configuration value.

| Value | Description                            |
|-------|----------------------------------------|
| 0     | No limit on the size of header fields. |



'WJpV2' must be used for request headers exceeding 10,000 bytes.

The following is an example of how to configure `limit_request_header_field_size`.

```
"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads": 8,
  "limit_request_header_field_size": 100
  ...
}
```

When `limit_request_header_field_size` is set to 100, the following message is displayed if the size of each header field exceeds 100.

\* Closing connection 0  
The size of a request header field exceeds the server limit.

## limit\_request\_line\_size

Specifies the maximum size of a client request's HTTP line.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | bytes       |
| Range         | 0 ~ 16382   |
| Default Value | 8190        |

The following describes each configuration value.

| Value | Description                                 |
|-------|---------------------------------------------|
| 0     | No limit on the length of the request line. |



'WJPv2' must be used for request lines exceeding 10,000 bytes.

The following is an example of how to configure limit\_request\_line\_size.

```
"node": {  
  "name": "webtob_node",  
  "hth_count": 1,  
  "worker_threads": 8,  
  "limit_request_line_size": 100  
  ...  
}
```

If limit\_request\_line\_size is set to 100, the following message is displayed when the length of the request line exceeds 100.

\* Closing connection 0  
Request Uri too Wrong

## system\_filters

Sets the filter to be applied when system events (ON\_START, ON\_STOP) occur.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

### 3.10.2. Example

The following is an example of configuring the NODE section.

```
"node": {
  "name": "webtob_node",
  "hth_count": 1,
  "worker_threads": 8,
  "hth_schedule": "RR",
  "connection_pool_size": 8192,
  "graceful_shutdown_timeout": 60,
  "listen_backlog": 4096,
  "cache_key": "HOST_URI",
  "cache_entry": 128,
  "max_cache_memory_size": 100,
  "cache_max_file_size": 8192,
  "limit_request_body_size": 0,
  "limit_request_header_field_count": 100,
  "limit_request_header_field_size": 8190,
  "limit_request_line_size": 8190,
  "system_filters" : [ filter1 ]
}
```

## 3.11. SERVER Section

In the SERVER section, you can configure connection information (Port, SSL, etc.) that allows WebtoB to receive client requests.

Each protocol that WebtoB can handle must be set in the SERVER section.

WebtoB provides SERVER types such as '**http**', '**wjp**', and '**admin**', each of which can be configured in detail.

### 3.11.1. Configuration Items

The following is the configuration format of the SERVER section.

```
"server": {
  "http": {
    #"common_config": {                                # COMMON
      #"doc_root": string,                               # "docs/", $ENV, R.PATH
      #"error_document": [string],
      #"enable_keepalive": boolean,                      # true
    }
```

```

        # "keepalive_timeout": integer,                # 60 (1-3600)
        # "keepalive_max": integer,                    # 0 (0-INT_MAX)
        # "keepalive_error_status_code": [integer],
        # "service_order": string,                      # "uri,ext"
        # "access_log": string,
        # "headers": [string],
        # "access": string,                             # "#default_server_access"
        # "enable_dos_block": boolean,                 # false
        # "dos_block_table_size": integer,              # 3097 (1-INT_MAX)
        # "dos_block_page_count": integer,              # 5 (0-INT_MAX)
        # "dos_block_page_interval": integer,           # 1 (1-INT_MAX)
        # "dos_block_site_count": integer,              # 50 (0-INT_MAX)
        # "dos_block_site_interval": integer,           # 1 (1-INT_MAX)
        # "dos_block_period": integer,                  # 30 (1-INT_MAX)
        # "dos_block_white_list": [string],
        # "enable_url_rewrite": boolean,                # false
        # "url_rewrite_config": string,
        # "alias": [string],
        # "index_name": string,                         # "index.html"
        # "filters": [string],
        # "rpaf_header": string,
        # "enable_directory_index": boolean,            # false
        # "enable_convert_url_charset": boolean,        # false
        # "convert_url_charset": {
            # "from": string,                          # "UTF-8"
            # "to": string,                             # "EUC-KR"
            # "except": [string]
        },
        # "expires": {
            # "expires_time": string,
            # "mimetype": string,
            # "url": string
        },
        # "enable_allow_header_on_405": boolean        # true
    },
    "http_servers": [
        {
            # "common_config": {...},                  # COMMON
            "name": string,
            # "port": [integer],                        # [80] | [443] (1-65535)
            # "listen_address": [string],
            # "enable_ssl": boolean,                    # false
            # "ssl_name": [string],
            # "idle_timeout": integer,                  # 300 (1-3600)
            # "initial_connection_timeout": integer,    # 10 (0-INT_MAX)
            # "request_header_timeout": integer,        # 60 (0-INT_MAX)
            # "request_body_timeout": integer,          # 0 (0-INT_MAX)
            # "request_process_timeout": integer,       # 300 (0-INT_MAX)
            # "request_process_timeout_status": integer, # 500 (511-599)
            # "enable_http2": boolean,                 # false
            # "bind_ipv6_only": boolean,               # false
            # "enable_expect_100_continue": boolean,   # true
            # "vhosts": [
                {
                    # "common_config": {...},          # COMMON
                    "name": string,
                    "host_name": [string]
                }
            ]
        }
    ]

```

```

    ]
  },
  # "mime_type_file": string,                # "mime.types", $ENV
  # "default_mime_type": string              # "text/html"
},
# "wjp": {
  "wjp_servers": [
    {
      "name": string,
      # "min_proc": integer,                # 1 (1-INT_MAX)
      # "max_proc": integer,
      # "svr_chk_time": integer,            # 60 (0-3600)
      # "flow_control": integer,            # 50 (1-INT_MAX)
      # "max_jengine_count": integer,       # 64 (1-INT_MAX)
      # "ping_timeout_status": integer      # 503 (511-599)
    }
  ],
  # "port": integer,                        # 9999 (1-65535)
  # "listen_address": [string],
  # "enable_ssl": boolean,                  # false
  # "ssl_name": string,
  # "access": string                        # "#default_server_access"
}

# "admin": {
  # "port": integer,                        # 9090 (1-65535)
  # "access_log": string
}

# "tcp": {
  "tcp_servers": [
    {
      "name": string,
      "server_address": [string],
      # "port": [integer],                  # (1-65535)
      # "listen_address": [string],
      # "idle_timeout": integer,             # 300 (0-INT_MAX)
      # "initial_connection_timeout",       # 10 (0-INT_MAX)
      # "access": string                    # "#default_server_access"
    }
  ]
}
}

```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the SERVER section configuration items.

## http (Required)

Sets the connection information for processing HTTP requests.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## http/common\_config

A common setting for HTTP servers. If set in the parent item, the settings will be reflected to the child items without the need for additional configuration.

| Item      | Description                                                                               |
|-----------|-------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                    |
| Priority  | The priority of the setting is as follows:<br><br>1. vhosts<br>2. http_servers<br>3. http |

## http/common\_config/doc\_root

Specifies the top-level path of the documents served by WebtoB.

Environment variables can be used when configuring this setting. If a relative path is set, it will be converted to an absolute path based on "\$WEBTOB\_HOME".

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "docs"               |

The following is an example of how to configure doc\_root:

```
"server": {  
  "http": {  
    "common_config": {  
      "doc_root": "docs",  
    }  
  }  
}
```

If doc\_root is set in both common\_config and http\_servers, the setting in http\_servers will take precedence over that in common\_config.

## http/common\_config/error\_document

Specifies the use of a user specified page instead of the HTTP error page. Configure the names in the ERRORDOCUMENT section. This setting has a lower priority than the error\_document in the DIRECTORY setting.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 64 items (within 255 characters) |

The following is an example of how to configure `error_document`. In this case, the name of the `error_document` must be written in string format, and the `error_document` must be specified in the `error_document` section.

```
"server": {
  "http": {
    "common_config": {
      "error_document": ["error1"]
    }
    ...
  "error_document": {
    "error_document_list": [
      {
        "name": "error1",
        "status": 404,
        "url": "/htmls/not_found.html"
      }
    ]
  }
},
```

## http/common\_config/enable\_keepalive

Sets whether to use Keepalive (HTTP persistent connection).

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | True        |

The following describes each configuration value.

| Value | Description                                                                       |
|-------|-----------------------------------------------------------------------------------|
| true  | Users can reuse connections to handle multiple requests with a single connection. |
| false | Users must reconnect the socket every time to process a request.                  |

The following is an example of how to configure `enable_keepalive`. To configure `keepalive_timeout` and `keepalive_max`, `enable_keepalive` must be set to true.

```
"server": {
  "http": {
    "common_config": {
      "enable_keepalive": true,
      "keepalive_timeout": 60,
```

```
"keepalive_max":0,  
},
```

### http/common\_config/keepalive\_timeout

Specifies the time period during which Keepalive remains active. After the user request has been processed, the connection is disconnected after the specified time.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ 3600    |
| Default Value | 60          |

The following is an example of how to configure keepalive\_timeout.

```
"server": {  
  "http": {  
    "common_config": {  
      "enable_keepalive": true,  
      "keepalive_timeout": 10,  
      "keepalive_max":0,  
    },  
  },  
}
```

The connection time can be checked through Idle\_time in ci of wsadmin. When called from a browser, the connection may be terminated before the keepalive\_timeout setting value because each browser has its own connection time policy.



```
[wsadmin]>> ci
```

```
-----  
-----  
| HTH-0 : Connection info  
|
```

```
-----  
-----  
| No | Server |   Local Address   | Remote Address | Remote Type | Ssl |  
Status | Request Count | Idle Time | Mapping No |  
-----  
-----  
| 5 | http1 | 192.168.0.1:80 | 192.168.0.51:50965 | CLIENT | No | READY  
| 0 | | 7 | -1 |  
-----  
-----
```

## http/common\_config/keepalive\_max

Specifies when to limit the Keepalive request count.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                         |
|-------|-------------------------------------|
| 0     | No limit on the number of requests. |

The following is an example of how to configure keepalive\_max:

```
"server": {  
  "http": {  
    "common_config": {  
      "enable_keepalive": true,  
      "keepalive_timeout": 60,  
      "keepalive_max": 2,  
    },  
  },  
}
```



The number of connections can be checked through Request Count in ci of wsadmin.

```
[wsadmin]>> ci  
-----  
| HTH-0 : Connection info  
|  
-----  
| No | Server | Local Address | Remote Address | Remote Type | Ssl |  
Status | Request Count | Idle Time | Mapping No |  
-----  
| 5 | http1 | 192.168.0.1:80 | 192.168.0.51:50965 | CLIENT | No | READY  
| 0 | | 7 | -1 |  
-----
```

## http/common\_config/keepalive\_error\_status\_code

Specifies the HTTP status code to use when the user connection is kept alive (persistent connection,

keep-alive) after sending an error response.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (integer)                        |
| Range     | Up to 64 items (300 - 303   305 - 599) |

Normally, if WebtoB sends an error response with status code of 3xx (excluding "304 Not Modified"), 4xx, or 5xx, the client connection is terminated. However, if WebtoB sends a status code specified in this option, the client connection is maintained.

The following is an example of WebtoB keeping the connection with client alive after responding with either a "302 Found" or "404 Not Found" message.

```
...
"server": {
  "http": {
    "common_config": {
      "enable_keepalive": true,
      "keepalive_timeout": 60,
      "keepalive_max": 0,
      "keepalive_error_status_code": [302, 404],
    },
  },
  ...
}
```

### http/common\_config/service\_order

Sets the priority between URI and EXT in the SERVICE section when determining the Destination to process the user request.

| Item          | Description           |
|---------------|-----------------------|
| Data Type     | String                |
| Range         | "uri,ext"   "ext,uri" |
| Default Value | "uri,ext"             |

The following describes each configuration value.

| Value   | Description                                                                                                           |
|---------|-----------------------------------------------------------------------------------------------------------------------|
| uri,ext | Checks the URI section configuration first. If the configuration does not exist, check the EXT section configuration. |
| ext,uri | Checks the EXT section configuration first. If the configuration does not exist, check the URI section configuration. |



If neither option is specified, the default HTML service (Worker Thread) processes the request.

The following is an example of how to configure `service_order`.

```
...
"server": {
  "http": {
    "common_config": {
      "service_order": : "uri,ext"
    },
  },
  ...
}
```

### http/common\_config/access\_log

Specifies the LOGGING section name corresponding to the access log.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

The following is an example of how to configure `access_log`. The `access_log` specified in the setting must be defined in the `access_log` entry of the logging section.

```
...
"server": {
  "http": {
    "common_config": {
      "access_log": "access_log1"
    },
  },
  ...
"logging": {
  "access_log": [
    {
      "name": "access_log1",
      "level": "INFO",
      "format": "DEFAULT",
      ...
    }
  ],
  ...
}
```

### http/common\_config/headers

Sets the name of the HEADERS section to apply.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

The following is an example of how to configure headers. The names of the headers are specified in string format in the `headers_list` of the headers section. Multiple headers can be set.

```

...
"server": {
  "http": {
    "common_config": {
      "headers": ["header1", "header2"]
    },
    ...
  "headers": {
    "headers_list": [
      {
        "name": "header1",
        "action": "AppendResponse",
        "field_name": "Content-Type",
        "field_value": "; charset=ISO"
      },
      {
        "name": "header2",
        "action": "AddRequest",
        "field_name": "Accept-Encoding",
        "field_value": "GZIP"
      },
    ],
  }
}

```

### http/common\_config/access

Sets whether to allow a request based on its method, whether specific headers are included, and the source IP address. The setting value must match the name set in [access/access\\_list/name](#).

| Item          | Description              |
|---------------|--------------------------|
| Data Type     | String                   |
| Range         | Up to 255 characters     |
| Default Value | "#default_server_access" |

The default value "#default\_server\_access" is automatically generated internally even if it is not explicitly defined in the [access/access\\_list/name](#) section.



For more information about "#default\_server\_access", refer to [ACCESS section - Default Settings](#).

The following is an example of how to configure access.

```

...
"server": {
  "http": {
    "common_config": {
      "access": "access1"
    },
    ...
  }
}

```

```

"access": {
  "access_list": [
    {
      "name": "access1",
      "policy": "blacklist",
      "deny_network": [ "192.168.1.43/255.255.255.0" ]
    }
  ]
}

```

## http/common\_config/enable\_dos\_block

Sets whether to use the DoS attack prevention feature.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following is an example of how to configure enable\_dos\_block. Set to true to enable the DoS attack prevention feature.

```

...
"server": {
  "http": {
    "common_config": {
      ...
      "enable_dos_block":true,
      "dos_block_table_size":3097,
      "dos_block_site_count":50,
      "dos_block_site_interval":1,
      "dos_block_period":30
      ...
    },
    ...
  },
  ...
}

```

## http/common\_config/dos\_block\_table\_size

Sets the maximum number of IP addresses to be considered as DoS attacks. Any IP addresses exceeding the set value will be blocked.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 3097        |



Since the table is managed by HTH, if the HTH value is set to greater than 1, the

number of IP addresses managed will be 'dos\_block\_table\_size × hth\_count'.

The following is an example of how to configure dos\_block\_table\_size. Incoming IPs are not managed individually but are managed as objects for each IP. For example, if 192.168.0.1 and 192.168.0.2 come in, each IP is managed as a separate object.

```
...
"server": {
  "http": {
    "common_config": {
      ...
      "enable_dos_block":true,
      "dos_block_table_size":3097,
      "dos_block_site_count":50,
      "dos_block_site_interval":1,
      "dos_block_period":30
      ...
    },
    ...
  },
  ...
}
```

### http/common\_config/dos\_block\_page\_count

Specifies the number of requests to the same page.

If a user IP address requests the same page more than the specified number of times within the dos\_block\_page\_interval, the IP address will be blocked during the time specified in the dos\_block\_period period.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 5           |

The following describes each configuration value.

| Value | Description                                      |
|-------|--------------------------------------------------|
| 0     | Does not check for DoS attacks on the same page. |

The following is an example of how to configure dos\_block\_page\_count. If more requests than the specified count (5) occur within the interval, the corresponding IP address is blocked. The dos\_block\_page\_count is used in conjunction with dos\_block\_page\_interval.

```
...
"server": {
  "http": {
    "common_config": {
      ...
    },
    ...
  },
  ...
}
```

```

        "enable_dos_block":true,
        "dos_block_table_size":1,
        "dos_block_page_count":5,
        "dos_block_page_interval":1,
        "dos_block_period":30,
        ...
    },
    ...

```

## http/common\_config/dos\_block\_page\_interval

Specifies the period to check for DoS attacks on the same page.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ INT_MAX |
| Default Value | 1           |

The following is an example of how to configure dos\_block\_page\_interval. If more requests than the specified count occur within the interval (100 seconds), the corresponding IP address is blocked. The dos\_block\_page\_interval is used in conjunction with dos\_block\_page\_count.

```

...
"server": {
  "http": {
    "common_config": {
      ...
      "enable_dos_block":true,
      "dos_block_table_size":1,
      "dos_block_page_count":5,
      "dos_block_page_interval":1,
      "dos_block_period":30,
      ...
    },
    ...
  }
}
...

```

## http/common\_config/dos\_block\_site\_count

Sets the number of requests for the entire site.

If a user's IP address request any page on the site more than the specified number of times within the dos\_block\_site\_interval, the IP address will be blocked during the time specified in dos\_block\_period.

| Item      | Description |
|-----------|-------------|
| Data Type | Integer     |

| Item          | Description |
|---------------|-------------|
| Range         | 1 ~ INT_MAX |
| Default Value | 50          |

The following describes each configuration value.

| Value | Description                                      |
|-------|--------------------------------------------------|
| 0     | Does not check for DoS attacks on site requests. |

### **http/common\_config/dos\_block\_site\_interval**

Specifies the period to check for DoS attacks on the same site.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ INT_MAX |
| Default Value | 1           |

### **http/common\_config/dos\_block\_period**

Sets the period of time for which specific user IP addresses are blocked.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ INT_MAX |
| Default Value | 30          |

### **http/common\_config/dos\_block\_white\_list**

Specifies the IPs to exclude from DoS attack prevention. If a Proxy server is used, specify the server IP.

| Item          | Description                             |
|---------------|-----------------------------------------|
| Data Type     | Array (string)                          |
| Range         | Up to 256 items (within 255 characters) |
| Default Value | 30                                      |

The following is an example of how to configure dos\_block. In this example, if more requests than

the specified count (3) occur for an IP address with table\_size (1) during the interval (100 seconds), the IP address will be blocked for the period (100 seconds). However, IP addresses in the white\_list are excluded from the DoS blocking management.

```
...
"server": {
  "http": {
    "common_config": {
      ...
      "enable_dos_block":true,
      "dos_block_table_size":1,
      "dos_block_site_count":3,
      "dos_block_site_interval":100,
      "dos_block_period":100,
      "dos_block_white_list":["192.168.0.1"],
      ...
    },
    ...
  },
  ...
}
```

### http/common\_config/enable\_url\_rewrite

Sets whether to enable the URL rewriting feature.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following is an example of how to configure enable\_url\_rewrite.

```
...
"server": {
  "http": {
    "common_config": {
      ...
      "enable_url_rewrite":true,
      ...
    },
    ...
  },
  ...
}
```

### http/common\_config/url\_rewrite\_config

Specifies the path to the configuration file for using the URL rewriting feature.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

The following is an example of how to configure the `url_rewrite_config` settings. It places a configuration file written in regular expressions in the given path. However, this is valid only when the `enable_url_rewrite` is set to `true`.

```
...
"server": {
  "http": {
    "common_config": {
      ...
      "url_rewrite_config": "/home/tmax/webtob6/config/rewrite9.conf",
      ...
    },
    ...
  },
  ...
}
```

### http/common\_config/alias

To change the URI of a specific request to `realpath`, set the name defined in the `ALIAS` section.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 64 items (within 255 characters) |

The following is an example of how to configure an alias. In this case, when `url (/external/path/)` is called, it will be called as `real_path (/home/tmax/webtob6/docs/)`.

```
...
"server": {
  "http": {
    "common_config": {
      ...
      "alias": {
        "alias_list": [
          {
            "name": "alias1",
            "url": "/external/path/",
            "real_path": "/home/tmax/webtob6/docs/"
          }
        ]
      }
    },
    ...
  },
  ...
}
```

### http/common\_config/index\_name

Specifies the index page name for a service directory request. For example, if the request from client is `"/somedir/"`, the path `"/somedir/index.html"` is serviced.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "index.html"         |

## http/common\_config/filters

Sets the filters to be applied when the following events occur during request processing.

AFTER\_SEND\_REQUEST, AFTER\_SEND\_RESPONSE, BEFORE\_SEND\_REQUEST, BEFORE\_SEND\_RESPONSE, RECEIVE\_REQUEST, RECEIVE\_RESPONSE

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 15 items (within 255 characters) |

## http/common\_config/rpaf\_header

Sets the remote IP that was changed by a proxy server to the IP of the host that sent the request.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

Adding rpaf\_header allows users to specify a header. If a header set with rpaf\_header is contained, a remote IP will be changed to the header IP.



The remote IP address specified can be checked in the access.log.

```
"rpaf_header" = "X-Forwarded-For"
```

## http/common\_config/enable\_directory\_index

Sets whether to use the Directory Index feature when the index\_name file does not exist for a directory request.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

## http/common\_config/enable\_convert\_url\_charset

Sets whether to convert the charset of the HTTP request URL path.

A charset is a combination of a Coded Character Set (CCS) and a Character Encoding Scheme (CES), and Korean characters typically use EUC-KR or UTF-8. If the charset used in the request URL path differs from that used by the server, the `convert_url_charset` setting is used to convert the charset.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

## http/common\_config/convert\_url\_charset

Sets the feature for converting the charset of the HTTP request URL path.

| Item          | Description |
|---------------|-------------|
| Data Type     | Object      |
| Default Value | False       |

The following is an example that converts a URL encoded in UTF-8 to EUC-KR, while excluding requests forwarded to JEUS from conversion.

```
"convert_url_charset": {  
  "from": "UTF-8",  
  "to": "EUC-KR"  
  "except": ["JEUS"]  
}
```

## http/common\_config/convert\_url\_charset/from

Sets the charset used by the HTTP request URL path.

| Item          | Description        |
|---------------|--------------------|
| Data Type     | String             |
| Range         | "EUC-KR"   "UTF-8" |
| Default Value | "UTF-8"            |



Only "EUC-KR" and "UTF-8" are supported.

## http/common\_config/convert\_url\_charset/to

Sets the charset used by the server.

| Item          | Description        |
|---------------|--------------------|
| Data Type     | String             |
| Range         | "EUC-KR"   "UTF-8" |
| Default Value | "EUC-KR"           |



Only "EUC-KR" and "UTF-8" are supported.

## http/common\_config/convert\_url\_charset/except

Specifies the destination to be excluded from the convert\_url\_charset setting.

When a request is sent to the specified destination, the URI is not converted.

| Item      | Description                       |
|-----------|-----------------------------------|
| Data Type | Array (string)                    |
| Range     | "JEUS"   "REVERSE_PROXY"   "FCGI" |

## http/common\_config/expires

Sets the server response header information sent in response to the client request. This setting defines the expiry date and related headers for documents of specific MIME types.

When a client's browser reconnects to the same website, if a web document has been cached, it requests only the modified documents. In this case, the web browser sends a request including the specific MIME type and expiration-related information. The server verifies this information and returns only the expired documents.

The web server references the configuration file and sends the expiry date to the client.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |
| Range     | Up to 64 items |

The following is an example of how to configure the expires setting.

```
"expires": [  
  {  
    "mimetype": "text/html",  
    "expires_time": "604800s"  
  },  
  ...  
]
```

```

{
  "mimetype": "image/gif",
  "expires_time": "2419200s"
},
{
  "expires_time": "86400s"
},
{
  "url": "/news/",
  "expires_time": "1d"
},
{
  "mimetype": "text/html",
  "expires_time": "1w"
},
{
  "url": "/image/",
  "mimetype": "image/gif",
  "expires_time": "1M"
}
]

```

### http/common\_config/expires/expires\_time

Sets the expiration date. If only ExpiresTime is set, the expiration date is applied to the request header for all requests.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

The relative time format (<num><time intervals>) consists of the following components:

| Item             | Description                                                                                                                                                                                                                                                               |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <num>            | Integer value.                                                                                                                                                                                                                                                            |
| <time intervals> | <p>A suffix representing the following time units:</p> <ul style="list-style-type: none"> <li>◦ s (seconds)</li> <li>◦ m (minutes)</li> <li>◦ h (hours)</li> <li>◦ d (days)</li> <li>◦ w (weeks)</li> <li>◦ M (months, 30 days)</li> <li>◦ y (years, 365 days)</li> </ul> |

## http/common\_config/expires/mimetype

Sets the expiration date based on the response MIME type.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 127 characters |

## http/common\_config/expires/url

If the request URL matches, the corresponding expiration date is applied. The setting value must start with `/`.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## http/common\_config/enable\_allow\_header\_on\_405

Specifies whether to include the Allow header in a 405 Method Not Allowed response.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | True        |

The following describes each configuration value.

| Item  | Description                                         |
|-------|-----------------------------------------------------|
| true  | Includes the Allow header in 405 responses.         |
| false | Does not include the Allow header in 403 responses. |

## http/http\_servers (Required)

Sets the connection information for the servers that handle HTTP requests.

| Item      | Description     |
|-----------|-----------------|
| Data Type | Array (object)  |
| Range     | Up to 100 items |

## http/http\_servers/common\_config

A common setting for HTTP servers. If set in the parent item, the settings will be reflected to the child items without the need for additional configuration.

| Item      | Description                                                                               |
|-----------|-------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                    |
| Priority  | The priority of the setting is as follows:<br><br>1. vhosts<br>2. http_servers<br>3. http |

## http/http\_servers/name (Required)

Specifies the name of the HTTP server.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

## http/http\_servers/port

Specifies the service port of the HTTP server that can be accessed by the user.



Ports 1 to 1023 require root privileges. When using a regular user account, ports 1024 and higher must be used.

For more information, refer to [Process Privileges and Privileged Ports](#).

| Item          | Description                                           |
|---------------|-------------------------------------------------------|
| Data Type     | Array (integer)                                       |
| Range         | Up to 100 items (1 ~ 65535)                           |
| Default Value | 80 or 443 (If SSL is used, the default value is 443.) |



When both port and listen\_address are set, the port setting is ignored.

## http/http\_servers/listen\_address

Specifies the IP address and port number of the server that users can access. To open the service

port only for a specific IP address, configure it as "<IP>:<Port>".

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 100 items (within 53 characters) |



When both port and listen\_address are set, the port setting is ignored.

## http/http\_servers/enable\_ssl

Specifies whether to use the SSL/TLS protocol to connect to HTTP.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

## http/http\_servers/ssl\_name

Sets the SSL section items to apply. This applies only when enable\_ssl is set to true.

| Item      | Description                           |
|-----------|---------------------------------------|
| Data Type | Array (string)                        |
| Range     | Up to 64 items (within 31 characters) |

The following is an example of how to configure the ssl\_name:

```
...
"server": {
  "http": {
    "http_servers": {
      ...
      "enable_ssl": true,
      "ssl_name": ["ssl1"]
      ...
    },
    ...
  }
}
```

## http/http\_servers/idle\_timeout

Specifies the timeout to read/write data to/from a user socket.

When processing a user request, the socket is closed if the user does not write data to or read data from the socket during the specified time.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 1 ~ 3600    |
| Default Value | 300         |

### **http/http\_servers/initial\_connection\_timeout**

Specifies the timeout for receiving the first request from the user.

This setting closes the socket if the user does not send any requests within the specified time after establishing the TCP connection. This also applies to SSL connections. The socket will be closed if the user does not complete SSL connection within the timeout period or fails to send a request after the connection is established.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 0 ~ INT_MAX |
| Default Value | 10          |

The following describes each configuration value.

| Value | Description                   |
|-------|-------------------------------|
| 0     | No timeout period is applied. |

### **http/http\_servers/request\_header\_timeout**

Specifies the amount of time to wait for HTTP request headers.

This setting applies while a user sends HTTP request headers. If all request headers are not received within the specified time, a '408 Request Timeout' response is returned.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 0 ~ INT_MAX |
| Default Value | 60          |

The following describes each configuration value.

| Value | Description                                     |
|-------|-------------------------------------------------|
| 0     | No time limit for sending HTTP request headers. |

## http/http\_servers/request\_body\_timeout

Specifies the amount of time to wait for the HTTP request body, if present.

This setting applies while a user sends the HTTP request body. If the entire request body is not received within the specified time, a '408 Request Timeout' response is returned.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                                      |
|-------|--------------------------------------------------|
| 0     | No time limit for sending the HTTP request body. |

## http/http\_servers/request\_process\_timeout

Specifies the maximum time to wait after parsing an HTTP request before returning a response.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 0 ~ INT_MAX |
| Default Value | 300         |



Approximately 72 bytes of memory are used per HTTP request during the specified time. Therefore, setting a longer time may increase memory usage time per request and cause system overload.

## http/http\_servers/request\_process\_timeout\_status

If an HTTP request processing timeout occurs, the server returns the HTTP status code specified by this setting.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 511 ~ 599   |
| Default Value | 500         |

### **http/http\_servers/enable\_http2**

Sets whether HTTP/2 connections and requests are allowed.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **http/http\_servers/bind\_ipv6\_only**

Sets whether the server allows only IPv6 addresses. If set to false, the server allows both IPv4 and IPv6.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **http/http\_servers/enable\_expect\_100\_continue**

Sets whether a 100 Continue response is returned when the HTTP request header contains "Expect: 100-continue".

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | True        |

The following describes each configuration value.

| Value | Description                                                                                           |
|-------|-------------------------------------------------------------------------------------------------------|
| true  | Returns a 100 Continue response when the HTTP request header contains "Expect: 100-continue".         |
| false | Does not return a 100 Continue response when the HTTP request header contains "Expect: 100-continue". |

## http/http\_servers/vhosts

Configures the virtual host feature that allows the HTTP server to provide different services depending on the requested address.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |
| Range     | Up to 64 items |

The following is an example vhosts configuration:

```
...
"server": {
  "http": {
    "http_servers": {
      ...
      "vhosts": [
        {
          "name": "vhost1",
          "host_name": ["www.vh1.com"],
          "common_config": {...}
        },
        {
          "name": "vhost2",
          "host_name": ["www.vh2.com"],
          "common_config": {...}
        }
      ],
    }
  }
  ...
},
...
```

### http/http\_servers/vhosts/name (Required)

Sets the name of the virtual host.

| Item      | Description    |
|-----------|----------------|
| Data Type | String         |
| Range     | Up to 64 items |

### http/http\_servers/vhosts/host\_name (Required)

Specifies the host name used to access the virtual host. Each host name must be configured differently to distinguish between virtual hosts.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 10 items (within 127 characters) |

## http/mime\_type\_file

Specifies the path to the MIME type configuration file that maps MIME types to file extensions. Environment variables can be used when configuring this setting.

This path can be specified as an absolute path or a relative path to \$WEBTOB\_HOME. If no value is set, the feature is disabled.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 255 characters |
| Default Value | "mime.types"         |

## http/default\_mime\_type

Specifies the default Content-Type for documents whose MIME type cannot be determined. This setting has a lower priority than the default\_mime\_type in the DIRECTORY setting.

| Item          | Description          |
|---------------|----------------------|
| Data Type     | String               |
| Range         | Up to 127 characters |
| Default Value | "text/html"          |

The following is an example of how to configure mime\_type:

```
...
"server": {
  "http": {
    "http_servers": {
      ...
      "mime_type_file": "mime.types",
      "default_mime_type": "text/html"
      ...
    },
    ...
  },
  ...
}
```



Since ext has a higher priority than mime\_type, you need to remove the ext setting to configure mime\_type.

## wjp

Sets WJP's connection information for communication with JEUS.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## wjp/port

Sets the service port required for integrating WebtoB with JEUS.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ 65535   |
| Default Value | 9999        |



The additional ports used are calculated as  $(/node/process\_count) * (/node/htb\_count)$ .

For example, if the process\_count is set to 3 and the htb\_count is set to 2, and the port is set to 9900, each HTH will use ports in the range 9900 to 9905

If WebtoB opens the port, JEUS connects to the port using the following configuration in the 'domain.xml'.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<domain version="8.0" xmlns="http://www.tmaxsoft.com/xml/ns/jeus">
  ...
  <servers>
    <server>
      <web-engine>
        <web-connections>
          <webtob-connector>
            <name>webtob1</name>
            <wjp-version>2</wjp-version>
            <registration-id>MyGroup</registration-id>
            <network-address>
              <port>9999</port>
            </network-address>
            <thread-pool>
              <number>10</number>
            </thread-pool>
          </webtob-connector>
        </web-connections>
      </web-engine>
    </server>
  </servers>
  ...
</domain>
```



From WebtoB 6 onward, WJP version 1 is not supported. Therefore, ensure that the `<wjp-version>` value is not set to 1 in JEUS's domain.xml.

### wjp/listen\_address

Specifies the IP address of the server that users can access. To open the JEUS port only for a specific IP address, configure it as "`<IP>`".

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 100 items (within 53 characters) |

### wjp/enable\_ssl

Specifies the option to use the SSL/TLS protocol for service port.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |



You can set Truststore generated by WebtoB SSL in JEUS to establish a connection.

### wjp/ssl\_name

Sets the SSL section items to apply. This applies only when `enable_ssl` is set to true.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### wjp/access

Sets whether to allow connection attempts from the client IP address. In this case, the setting value must match the name set in `access/access_list/name`.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

| Item    | Description              |
|---------|--------------------------|
| Default | "#default_server_access" |

The default value "#default\_server\_access" is automatically generated internally even if it is not explicitly defined in the [access/access\\_list/name](#) section.



For more information about "#default\_server\_access", refer to [ACCESS section - Default Settings](#).

The following is an example of access settings.

```
...
"server": {
  "wjp": {
    "access": "access1"
  },
  ...
"access": {
  "access_list": [
    {
      "name": "access1",
      "policy": "blacklist",
      "deny_network": [ "192.168.1.43/255.255.255.0" ]
    }
  ]
}
}
```

## wjp/wjp\_servers (Required)

Set the connection information for the servers that will process WJP.

| Item      | Description     |
|-----------|-----------------|
| Data Type | Array (object)  |
| Range     | Up to 100 items |

The following is an example of how to configure wjp\_servers:

```
...
"server": {
  "wjp": {
    "wjp_servers": [
      {
        "name": "JeusServer1",
        "svr_chk_time": 60
      },
      {
        "name": "JeusServer2",
        "svr_chk_time": 30
      }
    ]
  }
}
```

```
}  
]  
...
```

### **wjp/wjp\_servers/name (Required)**

Sets the name of the WJP server.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 15 characters |

### **wjp/wjp\_servers/svr\_chk\_time**

Specifies the interval at which the connection with JEUS is checked.

For a connection in Ready state without service requests, if there is no response to KeepAlive requests for two consecutive svr\_chk\_time periods, the connection is recognized as abnormal and is terminated and excluded from service distribution.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Unit          | seconds     |
| Range         | 0 ~ 3600    |
| Default Value | 60          |

### **wjp/wjp\_servers/flow\_control**

Sets the maximum size of the response buffer to control the amount of responses received from the server. A smaller value will cause slower page loading, while larger values results in faster responses.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 50          |

### **wjp/wjp\_servers/max\_jengine\_count**

Sets the maximum number of JEUS servers that can be connected to one server.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ INT_MAX |
| Default Value | 64          |

## wjp/wjp\_servers/ping\_timeout\_status

Sets the HTTP status code to be returned if a PING request to the JSV server times out.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 511 ~ 599   |
| Default Value | 503         |

## admin

Sets the connection information for the WebtoB Admin server. This setting is used when calling the WebtoB API provided by the Admin server.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## admin/port

Sets the port number of the admin server.



The configured port number can be checked in the logs when wsadmin is run.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 1 ~ 65535   |
| Default Value | 9090        |



The additional ports used are calculated as `/node/process_count`.

For example, if the process\_count is set to 3 and the port is set to 9090, each process will use ports in the range 9090 to 9092.

## admin/access\_log

Specifies the LOGGING section name corresponding to the access log.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## tcp

Sets the TCP connection information. The TCP server acts as a proxy that forwards TCP connections from a specific port or IP address to another server. The TCP server never interprets transferred data.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## tcp/tcp\_servers (Required)

Sets connection information for servers to process TCP requests.

| Item      | Description     |
|-----------|-----------------|
| Data Type | Array (object)  |
| Range     | Up to 100 items |

## tcp/tcp\_servers/name (Required)

Sets the name of the TCP server.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

## tcp/tcp\_servers/server\_address (Required)

Sets the IP address and port number of the server that will handle client requests.

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 100 items (within 31 characters) |

### tcp/tcp\_servers/port (Required)

Specifies the service port of the TCP server that can be accessed by the user.

| Item      | Description                 |
|-----------|-----------------------------|
| Data Type | Array (integer)             |
| Range     | Up to 100 items (1 ~ 65535) |



When both port and listen\_address are set, the port setting is ignored.

### tcp/tcp\_servers/listen\_address

Specifies the IP address and port number of the server that users can access. To open the service port only for a specific IP address, configure it as "<IP>:<Port>".

| Item      | Description                            |
|-----------|----------------------------------------|
| Data Type | Array (string)                         |
| Range     | Up to 100 items (within 53 characters) |



When both port and listen\_address are set, the port setting is ignored.

### tcp/tcp\_servers/idle\_timeout

Specifies the timeout to read/write data to/from a user socket.

When processing a user request, the socket is closed if the user does not write data to or read data from the socket during the specified time.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 300         |

The following describes each configuration value.

| Value | Description                   |
|-------|-------------------------------|
| 0     | No time limit for idle state. |

## tcp/tcp\_servers/initial\_connection\_timeout

Specifies the timeout for receiving the first request from the user.

This setting closes the socket if the user does not send any requests within the specified time after establishing the TCP connection.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 10          |

The following describes each configuration value.

| Value | Description                   |
|-------|-------------------------------|
| 0     | No timeout period is applied. |

## tcp/tcp\_servers/access

Sets whether to allow connection attempts from the client IP address. In this case, the setting value must match the name set in [access/access\\_list/name](#)

| Item          | Description              |
|---------------|--------------------------|
| Data Type     | String                   |
| Range         | Up to 255 characters     |
| Default Value | "#default_server_access" |

The default value "#default\_server\_access" is automatically generated internally even if it is not explicitly defined in the [access/access\\_list/name](#) section.



For more information about "#default\_server\_access", refer to [ACCESS section - Default Settings](#).

The following is an example of access settings.

```
...
"server": {
  "tcp": {
    "tcp_servers": [
      {
        "name": "tcp1",
        "port": [5000],
        "server_address": [
          "192.168.1.20:5000"
        ],

```

```

        "access": "access1"
    }
]
}
...
"access": {
    "access_list": [
        {
            "name": "access1",
            "policy": "blacklist",
            "deny_network": [ "192.168.1.43/255.255.255.0" ]
        }
    ]
}
}

```

### 3.11.2. Example

The following is an example of configuring the SERVER section.

```

"server": {
    "http": {
        "common_config": {
            "doc_root": "docs",
            "idle_timeout": 300,
            "service_order": "uri,ext",
            "access_log": "access_log1",
            "keepalive": true,
            "keepalive_timeout": 60
        },
        "http_servers": [
            {
                "name": "http1",
                "port": [8080],
                "enable_ssl": false
            }
        ]
    }

    "wjp": {
        "port": 9999,
        "wjp_servers": [
            {
                "name": "MyGroup1",
                "svr_chk_time": 60
            }
        ]
    }

    "tcp": {
        "tcp_servers": [
            {
                "name": "tcp1",
                "port": [5000],
                "server_address": [
                    "192.168.1.20:5000"
                ]
            }
        ]
    }
}

```

```

    }
  ]
}

```

## 3.12. SERVICE section

Configures the internal server to handle HTTP requests based on the request's URI and EXT.

### 3.12.1. Configuration Items

The following is the configuration format of the SERVICE section.

```

"service": {
  "uri": [
    {
      "name": string,
      #"target_http_servers": [string],      # ["*"]
      "match": {
        #"rewrite": string,
        #"type": string,                    # "prefix"
        "target": string,
        #"redirect": string,
        #"redirect_status": integer,        # 302
        #"enable_cache": boolean,          # false
        #"ext": [string],                   # []
        #"goto_ext": boolean,               # false
        #"stop_if_no_ext": boolean          # true
      },
      "destination": {
        "type": string,
        "target": string
      },
      #"access": string
    }
  ]
  "ext": [
    {
      "name": string,
      #"target_http_servers": [string],      # ["*"]
      "match": {
        #"type": string,                    # "exact"
        "target": string,
        #"enable_cache": boolean            # false
      },
      "destination": {
        "type": string,
        "target": string
      },
      #"access": string
    }
  ]
}

```

```
}
```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the SERVICE section configuration items.

## uri

The URI section is used to provide different destinations to process requests according to the Uniform Resource Identifier (URI) value of the client request.

For example, if the address `http://www.tmax.co.kr/JSV/test` is requested, the `"/JSV/"` URI can be set to JEUS.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |



In the following example, the URI section must be configured for `<first>` or a URI that starts with `<first>` must be used.

```
http://<hostname>:<port>/<first>/<second>/index.html
```

## uri/name (Required)

Sets the name of the URI section.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

## uri/target\_http\_servers

Specifies the server name to process the request. (Example: `"target_http_servers": "http1"`)

If a specific vhost is to handle the request, configure it as 'server name.vhost name'. (Example: `"target_http_servers": "http1.vhost1"`)

| Item      | Description                           |
|-----------|---------------------------------------|
| Data Type | Array (string)                        |
| Range     | Up to 64 items (within 31 characters) |

| Item          | Description |
|---------------|-------------|
| Default Value | ["*"]       |

The following describes each configuration value.

| Value | Description             |
|-------|-------------------------|
| "*"   | Applies to all servers. |

## uri/match (Required)

Sets a rule to handle a request if its URI matches a specific pattern.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## uri/match/rewrite

Sets the value to change for uri/match/target.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |



This cannot be set if `/uri/match/type` is regex.

## uri/match/type

Specifies the type of pattern configured for the URI. The matching method used for HTTP Request path varies depending on the pattern type.

| Item          | Description                |
|---------------|----------------------------|
| Data Type     | String                     |
| Range         | "prefix", "exact", "regex" |
| Default Value | "prefix"                   |

The following describes each configuration value.

| Value  | Description                                                                                                                                                                                             |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| prefix | Matches if the pattern set in uri/match/target is the prefix of the HTTP request URL.<br>(Example: The pattern "/uri/" matches request paths such as "/uri/a/", "/uri/a/b/", and "/uri/a/b/c".)         |
| exact  | Matches if the pattern set in uri/match/target is the same as the HTTP request URL.<br>(Example: The pattern "/uri/" matches only when the request path is "/uri/". No other request paths will match.) |
| regex  | Matches if the HTTP request URL satisfies the regular expression pattern set in uri/match/target.<br>(Example: The pattern "/uri.*" matches request paths such as "/uri", "/uri/", and "/uri1".)        |

### uri/match/target (Required)

Specifies the pattern that matches the HTTP request path. Once matched, the URI section configuration is applied to the request.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### uri/match/redirect

Maps a request for the specified URI to another URI.

Depending on the value of redirect\_status set in the URI section, the value set in redirect is set in the Location header field of the HTTP response and delivered to the user. If the value of redirect\_status is omitted, the default value of "302 Found" is used.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### uri/match/redirect\_status

Specifies the HTTP status code to be used when using redirection.

| Item      | Description                       |
|-----------|-----------------------------------|
| Data Type | Integer                           |
| Range     | 301   302   303   305   407   410 |

| Item          | Description |
|---------------|-------------|
| Default Value | 302         |

The following describes each configuration value.

| Value | Alias     | Description                             |
|-------|-----------|-----------------------------------------|
| 301   | permanent | Responds with "301 Moved Permanently".  |
| 302   | found     | Responds with "302 Found".              |
| 303   | seeother  | Responds with "303 See Other".          |
| 305   | useproxy  | Responds with "305 Use Proxy".          |
| 307   | temp      | Responds with "307 Temporary Redirect". |
| 410   | gone      | Responds with "410 Gone".               |

### uri/match/enable\_cache

Sets whether the content is cached.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### uri/match/ext

Specifies additional extensions for URI matching. This is used when applying a URI section only to specific extensions under a particular path.

| Item          | Description    |
|---------------|----------------|
| Data Type     | Array (string) |
| Default Value | []             |

### uri/match/goto\_ext

Sets whether to perform EXT matching after URI matching.



When an EXT match occurs, the corresponding EXT service takes precedence, and the previously matched URI settings will not be applied.

| Item      | Description |
|-----------|-------------|
| Data Type | Boolean     |

| Item          | Description |
|---------------|-------------|
| Default Value | False       |

### **uri/match/stop\_if\_no\_ext**

When the goto\_ext setting is enabled and no matching EXT is found, the matching stops, and the system determines whether to apply the current URI service or continue searching for the next URI service.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | True        |

The following describes each configuration value.

| Value | Description                                                                               |
|-------|-------------------------------------------------------------------------------------------|
| true  | If no matching EXT found, matching stops and the current URI configuration is used.       |
| false | If no matching EXT found, the system continues searching for the next URI configuration . |

### **uri/destination (Required)**

Sets the destination of the URI service.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

### **uri/destination/type (Required)**

Sets the type of the destination.

| Item      | Description                                 |
|-----------|---------------------------------------------|
| Data Type | String                                      |
| Range     | "HTMLS"   "JEUS"   "REVERSE_PROXY"   "FCGI" |

### **uri/destination/target (Required)**

Sets the target of the destination. It must match the name specified in the DESTINATION section.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 127 characters |

## uri/match/access

Sets whether to allow a request based on its method, whether specific headers are included, and the source IP address. The setting value must match the name set in [access/access\\_list/name](#).

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## ext

Maps a client request to a specific process based on the requested file extension.

| Item      | Description    |
|-----------|----------------|
| Data Type | Array (object) |



WebtoB is already configured with mappings between basic MIME types and processes. However, if additional settings are needed, they can be modified in the relevant section.

## ext/name (Required)

Sets the name of the EXT section.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

## ext/target\_http\_servers

Specifies the server name to process the request. (Example: "target\_http\_servers": "http1")

If a specific vhost is to handle the request, configure it as 'server name.vhost name'. (Example: "target\_http\_servers": "http1.vhost1")

| Item          | Description                           |
|---------------|---------------------------------------|
| Data Type     | Array (string)                        |
| Range         | Up to 64 items (within 31 characters) |
| Default Value | ["*"]                                 |

The following describes each configuration value.

| Value | Description             |
|-------|-------------------------|
| "*"   | Applies to all servers. |

## ext/match (Required)

Sets a rule to process a file if its extension matches a specific pattern.

| Item      | Description |
|-----------|-------------|
| Data Type | Object      |

## ext/match/type

Sets the type of pattern configured in ext/match/target. The matching method used for HTTP Request path varies depending on the pattern type.

| Item          | Description                |
|---------------|----------------------------|
| Data Type     | String                     |
| Range         | "prefix", "exact", "regex" |
| Default Value | "exact"                    |

The following describes each configuration value.

| Value  | Description                                                                                                                                                                         |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| prefix | Matches if the pattern set in ext/match/target is the prefix of the extension. (Example: The pattern "abc" matches extensions such as "abc", "abc1", and "abc12".)                  |
| exact  | Matches if the set pattern matches the extension. (Example: The pattern "abc" matches only when the extension is "abc". No other extensions will match.)                            |
| regex  | Matches if the extension satisfies the regular expression pattern set in ext/match/target. (Example: The pattern "jsp[a-z]" matches extensions such as "jspa", "jspb", and "jspx".) |

### **ext/match/target (Required)**

Specifies the pattern that matches the HTTP request path. Once matched, the EXT section configuration is applied to the request.

| Item      | Description     |
|-----------|-----------------|
| Data Type | String          |
| Range     | Up to 127 items |

### **ext/match/enable\_cache**

Sets whether the content is cached.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### **ext/destination (Required)**

Sets the destination for the Extension service.

| Item      | Description |
|-----------|-------------|
| Data Type | String      |

### **ext/destination/type (Required)**

Sets the type of the destination.

| Item      | Description                                 |
|-----------|---------------------------------------------|
| Data Type | String                                      |
| Range     | "HTMLS"   "JEUS"   "REVERSE_PROXY"   "FCGI" |

### **ext/destination/target (Required)**

Sets the target of the destination. It must match the name specified in the DESTINATION section.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 127 characters |

## ext/match/access

Sets whether to allow a request based on its method, whether specific headers are included, and the source IP address. The setting value must match the name set in [access/access\\_list/name](#).

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### 3.12.2. Default Behavior

If a request does not match any settings in the SERVICE section, WebtoB routes the request to the default HTMLS named "#default\_htmls".



For more information about "#default\_htmls", refer to [DESTINATION Section - Default Settings](#).

### 3.12.3. Example

The following is an example of configuring the SERVICE section.

```
"service": {
  "uri": [
    {
      "name": "static_uri",
      "target_http_servers": [
        "*"
      ],
      "match": {
        "type": "prefix",
        "target": "/static",
        "rewrite": "/"
      },
      "destination": {
        "type": "HTMLS",
        "target": "htmls1"
      }
    }
  ],
  "ext": [
    {
      "name": "ext1",
      "target_http_servers": [
        "http1.vhost1"
      ],
      "match": {
        "type": "exact",
        "target": "text/html"
      },
    },
  ],
}
```

```

        "destination": {
            "type": "HTMLS",
            "target": "html1"
        }
    }
}

```

## 3.13. SSL Section

The SSL section configures SSL functions used in WebtoB. SSL service operates according to these settings.

### 3.13.1. Configuration Items

The following is the configuration format of the SSL section.

```

#"ssl": {
    #"common_config": {
        #"verify_depth": integer,
        #"protocols": [string],
        #"required_ciphers": string,
        #"tls13_required_ciphers": string
    },
    "TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256"
},
    #"ssl_configs": [
        {
            "name": string,
            "certificate_file": string,
            "certificate_key_file": string,
            #"certificate_chain_file": string,
            #"certificate_key_password": string,
            #"ca_certificate_file": string,
            #"ca_certificate_path": string,
            #"ssl_server_name": [string],
            #"verify_client": integer,
            #"renegotiation_level": string,
            #"enable_stapling": boolean,
            #"common_config": {...}
        }
    ],
    #"proxy_ssl_configs": [
        {
            "name": string,
            #"proxy_certificate_file": string,
            #"proxy_certificate_key_file": string,
            #"proxy_certificate_chain_file": string,
            #"proxy_certificate_key_password": string,
            #"proxy_ca_certificate_file": string,
            #"proxy_ca_certificate_path": string,
            #"ssl_server_name": [string],
            #"enable_insecure": boolean,
            #"common_config": {...}
        }
    ]
}

```

```
}  
  ]  
}
```



Refer to [Setting Value Formats and Configuration](#) for more information on symbols and details of the SSL section configuration items.

## common\_config

A common setting for SSL sections. If set in the parent item, the settings will be reflected to the child items without the need for additional configuration.

| Item      | Description                                                                                             |
|-----------|---------------------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                                  |
| Priority  | The priority of the setting is as follows:<br><br>1. "ssl_configs", "proxy_ssl_configs"<br><br>2. "ssl" |

## common\_config/verify\_depth

Specifies the level to trace and validate the chain of CAs for authentication. If verification from a single CA is sufficient, set this value to 1.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ INT_MAX |
| Default Value | 0           |

## common\_config/protocols

Specifies the protocols that can be used by the server. This can determine whether to support a specific TLS version. To disable a specific protocol, add a hyphen (-) before the protocol name.

| Item          | Description                                                |
|---------------|------------------------------------------------------------|
| Data Type     | Array (string)                                             |
| Range         | 1 to 4 items ("TLSv1"   "TLSv1.1"   "TLSv1.2"   "TLSv1.3") |
| Default Value | ["TLSv1", "TLSv1.1", "TLSv1.2", "TLSv1.3"]                 |



'SSLv2' and 'SSLv3' are not supported.

### **common\_config/required\_ciphers**

Specifies the ciphers that can be used by the server. You can also configure support for specific ciphers and SSL/TLS versions.

| Item          | Description           |
|---------------|-----------------------|
| Data Type     | String                |
| Range         | Up to 1023 characters |
| Default Value | "HIGH:!RSA"           |



Since WebtoB uses OpenSSL, refer to the OpenSSL guide for cipher names.

### **common\_config/tls13\_required\_ciphers**

Specifies the TLS 1.3 cipher suites that can be enabled. You can also configure support for specific ciphers and SSL/TLS versions.

| Item          | Description                                                                  |
|---------------|------------------------------------------------------------------------------|
| Data Type     | String                                                                       |
| Range         | Up to 1023 characters                                                        |
| Default Value | "TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256" |



Since WebtoB uses OpenSSL, refer to the OpenSSL guide for cipher names.

### **ssl\_configs**

Settings for when WebtoB operates as an SSL server.

| Item      | Description     |
|-----------|-----------------|
| Data Type | Array (object)  |
| Range     | Up to 100 items |

### **ssl\_configs/name (Required)**

The name of the SSL server configuration. You must set this 'name' when using SSL server configuration in other sections.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### **ssl\_configs/certificate\_file (Required)**

Specifies the server certificate encoded in PEM. Environment variables can be used when configuring this setting.

The certificate is encoded using DER rules and transmitted on the web in ASCII format. If the certificate is encrypted, you will be asked to enter a passphrase.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **ssl\_configs/certificate\_key\_file (Required)**

Specifies the private key of the PEM-encoded certificate used on the server. Environment variables can be used when configuring this setting.

If the key is not combined with the certificate, use this directive to specify the location of the key. Generally, this file is placed in the WebtoB SSL directory.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **ssl\_configs/certificate\_chain\_file**

Specifies the path to CA certificates used to create a server certificate chain. Environment variables can be used when configuring this setting.

For client authentication, the ca\_certificate\_file or ca\_certificate\_path item must be set.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## ssl\_configs/certificate\_key\_password

Specifies how to retrieve the passphrase for an encrypted private key file when SSL is used.

| Item          | Description |
|---------------|-------------|
| Data Type     | String      |
| Default Value | "builtin"   |

The following describes each configuration value.

| Value                       | Description                                                                                                                                                             |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| builtin                     | Asks for a passphrase when starting WebtoB.                                                                                                                             |
| exec:<program path>         | When starting WebtoB, a program is executed and its output is used as a passphrase. A file that is executed with 'exec' can be a compiled executable or a shell script. |
| raw:<passphrase>            | Uses this passphrase when WebtoB starts up.                                                                                                                             |
| file:<passphrase file path> | Uses the passphrase from this file that was generated by the mkpwd tool when WebtoB is started.                                                                         |

The following is an example of how to configure certificate\_key\_password.

```
"ssl_configs":[{
  "name":"ssl1",
  "certificate_file":"/home/webtob6/ssl/server.crt",
  "certificate_key_file":"/home/webtob6/ssl/server.key",
  "certificate_key_password":"exec:/home/webtob6/ssl/password.sh"
}],
```

## ssl\_configs/ca\_certificate\_file

Use this directive to verify the server from a single CA. The certificate file must be encoded in PEM. Environment variables can be used when configuring this setting.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

## ssl\_configs/ca\_certificate\_path

Specifies the directory where the certificate will be saved. Environment variables can be used when configuring this setting.

The certificate contains the information required for user authentication and should generally be

encoded in PEM format.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### **ssl\_configs/ssl\_server\_name**

Sets the server name that can be used as an alias in SSL.

| Item      | Description     |
|-----------|-----------------|
| Data Type | Array (string)  |
| Range     | Up to 100 items |

### **ssl\_configs/verify\_client**

Sets the authentication level to request from SSL clients.

| Item          | Description |
|---------------|-------------|
| Data Type     | Integer     |
| Range         | 0 ~ 3       |
| Default Value | 0           |

The following describes each configuration value.

| Value | Description                                                                                                                                  |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------|
| 0     | No authentication is requested.                                                                                                              |
| 1     | The user must provide valid authentication information to the server.                                                                        |
| 2     | The user must provide valid authentication information to the server.                                                                        |
| 3     | The user must provide valid authentication information. If the server does not have a certificate, the verification process is not required. |

### **ssl\_configs/renegotiation\_level**

Specifies the renegotiation level when SSL is used.

| Item          | Description                       |
|---------------|-----------------------------------|
| Data Type     | String                            |
| Range         | "secure"   "insecure"   "disable" |
| Default Value | "secure"                          |

The following describes each configuration value.

| Value    | Description                                                                                       |
|----------|---------------------------------------------------------------------------------------------------|
| secure   | Proceeds to renegotiate when the client and web server are secure. Example: RFC5746               |
| insecure | Proceeds to renegotiate although the client and web server are not secure. Example: CVE-2009-3555 |
| disable  | Does not renegotiate under any circumstances.                                                     |



If renegotiation proceeds in an insecure situation, it may be vulnerable to Man in the Middle (MITM) attacks or Denial of Service (DoS) attacks.

### ssl\_configs/enable\_stapling

Specifies whether WebtoB SSL operates with Online Certificate Status Protocol (OCSP) stapling.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

### ssl\_configs/common\_config

A common setting for SSL sections. If set in the parent item, the settings will be reflected to the child items without the need for additional configuration.

| Item      | Description                                                                                             |
|-----------|---------------------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                                  |
| Priority  | The priority of the setting is as follows:<br><br>1. "ssl_configs", "proxy_ssl_configs"<br><br>2. "ssl" |

### proxy\_ssl\_configs

Settings for when WebtoB operates as an SSL client. It is used when WebtoB acts as a reverse proxy and performs SSL communication.

| Item      | Description     |
|-----------|-----------------|
| Data Type | Array (object)  |
| Range     | Up to 100 items |

### proxy\_ssl\_configs/name (Required)

The name of the SSL client configuration. You must set this 'name' when using SSL in reverse proxy configuration.

| Item      | Description         |
|-----------|---------------------|
| Data Type | String              |
| Range     | Up to 31 characters |

### proxy\_ssl\_configs/enable\_insecure

Sets whether to allow SSL connections when the internal server's certificate is invalid.

| Item          | Description |
|---------------|-------------|
| Data Type     | Boolean     |
| Default Value | False       |

The following describes each configuration value.

| Value | Description             |
|-------|-------------------------|
| true  | Allows SSL connections. |

### proxy\_ssl\_configs/proxy\_certificate\_file

Specifies the client certificate file encoded in PEM. Environment variables can be used when configuring this setting. This must be configured if the internal server requires client authentication.

The certificate is encoded using DER rules and transmitted on the web in ASCII format. If the certificate is encrypted, you will be asked to enter a passphrase.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### proxy\_ssl\_configs/proxy\_certificate\_key\_file

Specifies the private key of the PEM-encoded certificate used for client authentication. Environment variables can be used when configuring this setting. This must be configured if the internal server requires client authentication.

If the key is not combined with the certificate, use this directive to specify the location of the key. Generally, this file is placed in the WebtoB SSL directory.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### proxy\_ssl\_configs/proxy\_certificate\_chain\_file

Specifies the path to CA certificates used to create a client certificate chain. Environment variables can be used when configuring this setting.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### proxy\_ssl\_configs/proxy\_certificate\_key\_password

Specifies how to retrieve the passphrase for an encrypted private key file when client authentication (proxy\_certificate\_file, proxy\_certificate\_key\_file) is used in PROXY\_SSL.

| Item          | Description |
|---------------|-------------|
| Data Type     | String      |
| Default Value | "builtin"   |

The following describes each configuration value.

| Value                       | Description                                                                                                                                                             |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| builtin                     | Asks for a passphrase when starting WebtoB.                                                                                                                             |
| exec:<program path>         | When starting WebtoB, a program is executed and its output is used as a passphrase. A file that is executed with 'exec' can be a compiled executable or a shell script. |
| raw:<passphrase>            | Uses this passphrase when WebtoB starts up.                                                                                                                             |
| file:<passphrase file path> | Uses the passphrase from this file that was generated by the mkpwd tool when WebtoB is started.                                                                         |

### proxy\_ssl\_configs/proxy\_ca\_certificate\_file

Use this directive to verify the server from a single CA. The certificate file must be encoded in PEM. Environment variables can be used when configuring this setting.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### proxy\_ssl\_configs/proxy\_ca\_certificate\_path

Sets the directory where the certificate will be stored. Environment variables can be used when configuring this setting.

The certificate contains the information required to verify the server's certificate and should generally be encoded in PEM format.

| Item      | Description          |
|-----------|----------------------|
| Data Type | String               |
| Range     | Up to 255 characters |

### proxy\_ssl\_configs/ssl\_server\_name

Sets the server name that can be used as an alias in SSL.

| Item      | Description                             |
|-----------|-----------------------------------------|
| Data Type | Array (string)                          |
| Range     | Up to 100 items (within 255 characters) |

### proxy\_ssl\_configs/common\_config

A common setting for SSL sections. If set in the parent item, the settings will be reflected to the child items without the need for additional configuration.

| Item      | Description                                                                                             |
|-----------|---------------------------------------------------------------------------------------------------------|
| Data Type | Object                                                                                                  |
| Priority  | The priority of the setting is as follows:<br><br>1. "ssl_configs", "proxy_ssl_configs"<br><br>2. "ssl" |

## 3.13.2. Example

The following is an example of configuring the SSL sections:

```
{
  "ssl": {
    "ssl_configs": [
      {
        "name": "ssl1",
        "certificate_file": "server.crt",
        "certificate_key_file": "server_key.crt",
        "certificate_chain_file": "server_chain.crt",
        "certificate_key_password": "builtin",
```

```

        "ca_certificate_file": "server_ca.crt",
        "ssl_server_name": [
            "example.com"
        ],
        "verify_client": 0,
        "renegotiation_level": "secure"
    }
],
"proxy_ssl_configs": [
    {
        "name": "proxy_ssl1",
        "proxy_certificate_file": "server.crt",
        "proxy_certificate_key_file": "server_key.crt",
        "proxy_certificate_chain_file": "server_chain.crt",
        "proxy_certificate_key_password": "builtin",
        "proxy_ca_certificate_file": "internal_server_ca.crt"
    }
]
}

```

## 4. WebtoB Process Structure

To ensure stable service, WebtoB automatically restarts processes when they terminate abnormally. By default, the Manager and Engine processes are started together.

### 4.1. Process Types

The following describes each process type of WebtoB.

| Process | Description                                        |
|---------|----------------------------------------------------|
| Manager | Restarts the abnormally terminated Engine process. |
| Engine  | Handles HTTP requests.                             |



The settings for the Manager and Engine must be identical.

If the settings of a running process differs from the new settings, you can choose whether to restart the process with the modified settings.

### 4.2. Manager Process

The Manager process manages the web server process (Engine) that handles HTTP requests. It monitors the Engine process and automatically restarts it if it terminates abnormally. If an Engine process is already running, the Manager process manages it.

Only one Manager process can run at a time. By default, WebtoB starts the Manager and Engine together, but the Manager can be started independently using the following command if necessary.

```
> webtob -a start-manager
```

In addition, WebtoB provides an Admin API via a Rest API. The Manager port is configured in the NODE section, and main APIs are as follows:

| API                              | Description                       |
|----------------------------------|-----------------------------------|
| <a href="#">/config</a>          | Views the configuration settings. |
| <a href="#">/shutdownmanager</a> | Shuts down the WebtoB Manager.    |

#### 4.2.1. Retrieving Configuration (/config)

Retrieves the configuration information of the currently running system. This includes not only the values specified in the configuration file, but also default values.

- Usage

```
{
  "path": string
}
```

| Option | Description                                                                                                                                                                                   |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| path   | Specifies the JSON path of the configuration file. This path must always start with a slash (/).<br>If the path is omitted or set to '/', all currently configuration settings are retrieved. |

- Example

The following is an example of retrieving the NODE section configuration using "/config" Admin API. For more information about configuration items of the NODE section, refer to [Configuration Items](#).

```
[Request]
{
  "path": "/node"
}

[Response]
{
  "result": {
    "cache_entry": 128,
    "cache_key": "HOST_URI",
    "cache_max_file_size": 8192,
    "connection_pool_size": 8192,
    "graceful_shutdown_timeout": 5,
    "hth_count": 1,
    "hth_schedule": "RR",
    "limit_request_body_size": 0,
    "limit_request_header_field_count": 100,
    "limit_request_header_field_size": 8190,
    "limit_request_line_size": 8190,
    "listen_backlog": 4096,
    "max_cache_memory_size": 100,
    "name": "webtob-server",
    "system_filters": [],
    "worker_threads": 1
  }
}
```

## 4.2.2. Shutting Down WebtoB Manager (/shutdownmanager)

Shuts down the running WebtoB Manager.

- Usage

```
{
  "timeout": integer
}
```

| Option  | Description                                                                                                                                                                                                                                   |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| timeout | Specifies the graceful shutdown timeout in seconds. If the WebtoB Manager does not shut down within the specified time, it is forcibly terminated. If omitted, the value set in <code>/node/graceful_shutdown_timeout</code> will be applied. |

- Example

The following is an example of calling the `/shutdownmanager` Admin API to forcibly terminate the WebtoB Manager if it does not shut down after a 30-second graceful shutdown attempt.

```
[Request]
{
  "timeout": 30
}

[Response]
{
  "result": {}
}
```



The response is returned immediately after WebtoB receives the command. However, the WebtoB Manager process may still be running even after the response is returned. Even if the WebtoB Manager is shut down, the Engine process is not terminated.

## 4.3. Engine Process

The Engine process handles HTTP requests. Each process has an index, starting from 0. The number of processes to run can be configured using `/node/process_count`.



Even while WebtoB is running, the number of Engine processes can be adjusted by modifying the `process_count` value and then restarting WebtoB.

By default, WebtoB starts the Manager and Engine together. However, if necessary, the Engine can be started independently using the following command.

```
> webtob -a start-engine
```

In addition, WebtoB provides an Admin API via a Rest API. For more information, refer to [WebtoB Admin API](#).



Specifying `%fullname%` or `%pid%` in the log file name creates separate log files for each process. If only `%name%` is set, log files are created only for the Manager and Engine processes.

## 4.4. Process Privileges and Privileged Ports

### 4.4.1. Privileged Ports and WebtoB Execution Permissions

Web servers typically use port 80 for HTTP and port 443 for HTTPS. These are classified as privileged ports (1-1023) by the operating system and can only be used by processes with root privileges.

Therefore, WebtoB must be run with root privileges to use these ports, and the environment variables must also be configured with root privileges.

The following example shows how to start WebtoB with root privileges to use privileged ports.

```
> sudo -E WEBTOB_HOME=/home/webtob/ ./webtob
```

### 4.4.2. Issues When Running as Root

When the WebtoB binary is executed with root privileges, both the Manager process and the Engine process are started with root privileges.

However, because the Engine process directly handles external HTTP requests, running it with root privileges increases security risks.

The following shows the process privileges when WebtoB binary is executed with root privileges.

```
> ps -eo pid,user,comm | grep WebtoB
396052 root    WebtoB Manager
396053 root    WebtoB Engine-0
396054 root    WebtoB Engine-1
396055 root    WebtoB Engine-2
```

### 4.4.3. Engine Privilege Separation

Using the `/node/user` setting, WebtoB can run the Engine process as a regular user. The specified user must have lower privileges than the user used to execute the WebtoB binary.

### Example of the `/node/user` setting

```
"node": {  
  "user": "webtob"  
}
```

After starting WebtoB with root privileges and specifying a regular user in `/node/user`, it behaves as follows:

- The Manager process runs with root privileges.
- The Engine process runs with the user privileges specified in `/node/user`.

This architecture enhances security by allowing the Manager to open privileged ports (1-1023) while the Engine operates with lower privileges.

The following shows the process privileges when the WebtoB binary is started with root privileges while `/node/user` is set to 'webtob'.

```
> ps -eo pid,user,comm | grep WebtoB  
396052 root    WebtoB Manager  
396053 webtob  WebtoB Engine-0  
396054 webtob  WebtoB Engine-1  
396055 webtob  WebtoB Engine-2
```



If the `/destination/jeus/enable_block_listen` is set to true, the Engine must also run with root privileges because it is responsible for reopening ports.

## 5. URL Rewrite

### 5.1. Configuring URLRewrite Feature

To use the URLRewrite feature, configuration related to Condition and Rule must be set in the 'url\_rewrite\_config' section of common\_config in /server/http.

The following is an example of the /server/http configuration to enable the URLRewrite feature:

```
"server": {  
  "http": {  
    "common_config": {  
      "url_rewrite_config": "config/rewrite.conf",  
      ...  
    }  
  }  
}
```

In url\_rewrite\_config, you can use RewriteCond and RewriteRule settings from Apache's mod\_rewrite feature. However, not all configurations are supported, and some functionalities are restricted.



The URLRewriteConfig settings described in the guide are based on Apache 2.2's 'mod\_rewrite'. Some of the described features may not function in WebtoB.

### 5.2. Defining Rewriting Conditions

To configure RewriteCond, specify rewriting conditions as follows. Compare the TestString and CondPattern. If the conditions are met, RewriteCond is changed to the pattern defined in RewriteRule configuration.

```
RewriteCond <TestString> <CondPattern> [flags]
```

#### 5.2.1. TestString Configuration

In TestString, the following reserved words and general strings can be used.

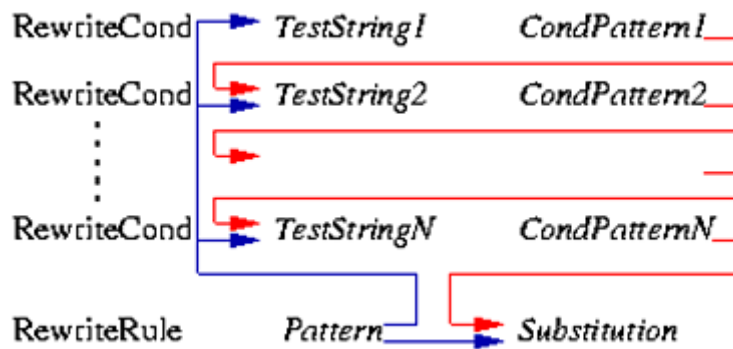
- **\$N (0 <= N <= 9)**

Refer to the pattern wrapped in parentheses among RewriteRule's patterns.

- **%N (1 <= N <= 9)**

Refer to the pattern wrapped in parentheses among RewriteCond's CondPatterns.

\$N and %N have the following Regex Back-Reference structure:



- **%{SERVER\_VARIABLE\_NAME}**

Refer to the additional variables among the server environment variables used in CGI.

| Type                        | Variable                                                                                                  |
|-----------------------------|-----------------------------------------------------------------------------------------------------------|
| Server Environment Variable | %{ENV:variable}: Refers to environment variables, %{HTTP:header}: Refers to HTTP request headers          |
| HTTP headers                | HTTP_USER_AGENT, HTTP_REFERER, HTTP_COOKIE, HTTP_FORWARDED, HTTP_HOST, HTTP_PROXY_CONNECTION, HTTP_ACCEPT |
| Connection & request        | REMOTE_ADDR, REMOTE_PORT, REMOTE_METHOD, QUERY_STRING, AUTH_TYPE                                          |
| Server internals            | DOCUMENT_ROOT, SERVER_NAME, SERVER_ADDR, SERVER_PORT, SERVER_PROTOCOL, SERVER_SOFTWARE                    |
| Date and time               | TIME_YEAR, TIME_MON, TIME_DAY, TIME_HOUR, TIME_SEC, TIME_WDAY, TIME                                       |
| Specials                    | THE_REQUEST, REQUEST_URI, HTTPS                                                                           |



The following variables provided by Apache's mod\_rewrite cannot be used.

```

%{SSL:variable}, %{LA-U:variable}, %{LA-F:variable}, REMOTE_HOST,
REMOTE_IDENT, SCRIPT_FILENAME, SCRIPT_USER, SCRIPT_GROUP, SERVER_ADMIN,
API_VERSION, REQUEST_FILENAME, IS_SUBREQ
  
```

## 5.2.2. CondPattern Configuration

CondPattern can use Perl compatible regular expressions. The following additional patterns can also be used.

| Directive    | Description                                                     |
|--------------|-----------------------------------------------------------------|
| !Pattern     | Specifies the cases that do not match the specified pattern(s). |
| <CondPattern | Matches when TestString is before CondPattern alphabetically.   |

| Directive    | Description                                                                                |
|--------------|--------------------------------------------------------------------------------------------|
| >CondPattern | Matches when TestString is after CondPattern alphabetically.                               |
| =CondPattern | Matches when TestString is same as CondPattern.                                            |
| -d           | Matches when TestString is considered a path and it is a directory.                        |
| -f           | Matches when TestString is considered a path and it is a file.                             |
| -s           | Matches when TestString is considered a path and the file size is greater than 0.          |
| -l           | Matches when TestString is considered a path and it is a symbolic link.                    |
| -x           | Matches when TestString is considered a path and it is an executable file.                 |
| -F           | Matches when TestString is considered a path and it is accessible in server configuration. |
| -U           | Matches when TestString is considered a URL and it is accessible in server configuration.  |



All of these patterns can be prefixed by an exclamation mark '!' to negate their meaning.

The following are the directives used in regular expressions.

| Directive | Description                                                                                              |
|-----------|----------------------------------------------------------------------------------------------------------|
| .         | Matches any single character.                                                                            |
| +         | Repeats the previous pattern one or more times.                                                          |
| *         | Repeats the previous pattern zero or more times.                                                         |
| ?         | Optionally matches the previous pattern.                                                                 |
| ^         | Optionally matches the previous pattern.                                                                 |
| \$        | Matches the end of the string.                                                                           |
| ( )       | Groups multiple characters into a single unit, and captures the match results for use in backreferences. |
| [ ]       | A character class - matches one of the characters.                                                       |
| [^ ]      | A negative character class - matches any character not specified.                                        |

### 5.2.3. Flags Configuration

You can change the way patterns are matched by using [flags] in CondPattern.

| Directive | Description                             |
|-----------|-----------------------------------------|
| nocase NC | Not case-sensitive in pattern matching. |

| Directive | Description                                                                                                                                                                           |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ornext OR | If a different RewriteCond exists after RewriteCond, combine the next RewriteCond with the logical OR operator. If not explicitly specified, the next condition is combined with AND. |
| novary NV | If HTTP Header is used as TestString, do not add Vary Header to Response.                                                                                                             |

The following is an example of using the [OR] flag:

```
RewriteCond %{HTTP_HOST} ^host1.* [OR]
RewriteCond %{HTTP_HOST} ^host2.* [OR]
RewriteCond %{HTTP_HOST} ^host3.*
RewriteRule ...
```

If the [OR] flag is not used, each RewriteCond/RewriteRule must be written separately.

## 5.3. Defining Rewriting Behavior

RewriteRule configures the rewriting of user requests. If a user request matches RewriteCond, the pattern in the user request is replaced with Substitution specified in the RewriteRule setting.

```
RewriteRule <Pattern> <Substitution> [flags]
```

### 5.3.1. Pattern Configuration

Patterns can use Perl compatible regular expressions, and you can use '!' to specify cases where the pattern does not match.



If '!' is used, the group pattern (\$N) enclosed in parentheses cannot be used because unmatched patterns do not have values for group patterns (\$N).

If RewriteRule is used alone, pattern matching uses URL-path. If RewriteRule is used with RewriteCond, the final matched pattern is used.

### 5.3.2. Substitution Configuration

Specifies what to substitute the matched URL with. The following values can be used.

| Value            | Description                                                                                                                                                                                                                                                     |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| file-system path | When specifying the absolute path of the file system that starts with '/', use the file in the user response. However, the configured path must exist in the file-system.                                                                                       |
| URL-path         | When a general URL path is used, use the appropriate resource.                                                                                                                                                                                                  |
| Absolute URL     | If an absolute URL is specified (for example, "http://<hostname>/file.html"), and the hostname matches the server, the scheme and hostname are stripped out and the resulting path is treated as a URL path. Otherwise, it is redirected to an external server. |
| -                | It means no substitution should be performed.                                                                                                                                                                                                                   |
| \$N (N=0..9)     | Indicates the Nth group pattern among RewriteRule's patterns.                                                                                                                                                                                                   |
| %N (N=1..9)      | Indicates the Nth group pattern among the last matched RewriteCond patterns.                                                                                                                                                                                    |
| %{VARIABLE}      | Refers to VARIABLE provided by the server. The items applied to TestString of RewriteCond can be used.                                                                                                                                                          |
| ?                | The Query string is not changed. In order to change the string, add '?' to the substitute string. In order to delete a query string, place a '?' at the end of the substitute string.                                                                           |



'\${mapname:key| default}' cannot be used because WebtoB does not support RewriteMap provided by Apache's mod\_rewrite.

### 5.3.3. Flags Configuration

The detailed behavior of RewriteRule can be configured through the settings in [flags].

Multiple flags can be set using commas (,), and the following values can be used:

- **B**

By default, backreferences used in Substitution (\$N or %N) remove the %-encoding of the URL. However, if the [B] option is used, the URL's %-encoding can be preserved.

For example, if set in the following, '/C%2b%2b' is mapped to 'index.php?show=/C++' by default. However, if the [B] option is used, it will be mapped to 'index.php?show=/C%2b%2b'.

```
RewriteRule ^(.*)$ index.php?show=$1
```

- **chain|C**

If the current rule does not match, the next rule will not be checked. If the next rule also uses the [C] option, it is also skipped.

- **cookie** | **CO=NAME:VAL:domain[:lifetime[:path[:secure[:httponly]]]]**

Add the Set-Cookie Header to a response to add a Cookie to a user's browser.

- **discardpathinfo** | **DPI**

In per-directory context, RewriteRule distinguishes URI and PATH\_INFO and combines them each time the rule is applied.

[DPI] options apply RewriteRule without distinguishing PATH\_INFO.

- **env** | **E=VAR:VAL**

Adds VAR=VAL to an environment variable. In VAL, regexp backreferences (\$N or %N) can be used. This environment variable can be used in SSI or CGI, and can be used as %{ENV:VAR} among RewriteCond patterns.

- **forbidden** | **F**

Sends a "403 Forbidden" response.

- **gone** | **G**

Sends a "410 Gone" response.

- **handle** | **H=Content-handler**

Configures a content-handler.

- **last** | **L**

Signifies the end of the rewriting process. This option is similar to the break command in C.

- **next** | **N**

Executes the rewriting process again from the start to the changed URL. This option is similar to the continue command in C.

Be careful when using this option, as it may cause an infinite loop.

- **nocase** | **NC**

Turns off case-sensitivity in patterns. For example, 'A' and 'a' are considered the same.

- **noescape** | **NE**

Does not use %-encoding for URL in the rewriting process.

- **nosubreq** | **NS**

Stops rewriting in the case of an internal request.

- **passthrough** | **PT**

Uses the result of rewriting from another handler.

- **qsappend|QSA**

When rewriting a query string, appends to the existing string instead of overwriting it.

- **redirect|R[=code]**

When the substitute string is an absolute URL, the URL is forcibly redirected even when the hostname matches a server's host.

If the code is not specified, 302 Moved Temporarily is used. In the code, the status code can be entered directly or temp (default), permanent, and seeother can be specified.

- **skip|S=num**

Skips the next 'num' rules. num is a numeric value.

- **type|T=MIME-type**

Specifies the content-type of the response.

## 5.4. Examples of URL Rewrite Configuration

The following are examples of URLRewriteConfig settings for different situations that may occur in a web server.

- **Example 1: Basic URL redirection**

The following is an example configuration that matches a URL pattern like 'www.test.com/' and redirects it to 'www.test.com/rewrite.html'.

```
# url rewrite config - ex1
RewriteCond %{HTTP_HOST} ^www\.test\.com$      # if {Host} == "www.test.com"
RewriteRule ^/$ /rewrite.html [L]              # then "/" > "/rewrite.html"
```

- **Example 2: Redirecting to an error page if a file does not exist**

The following is an example configuration that will convert a request like 'www.test.com/temp/xxx.html' to 'www.test.com/temp/temp\_error.html' when the file 'xxx.html' is not found in the temp directory.

```
# url rewrite config - ex2
RewriteCond %{REQUEST_FILENAME} !-f
# if {Requested file name} != file or the file pointer
RewriteRule ^/([^\s/]+)/$1/$1_error.html [L]
# then "/temp/xxx.html" > "/temp/temp_error.html"
```

- **Example 3: Redirecting HTTP requests to HTTPS**

The following is an example configuration that converts a request like 'http://www.test.com:80' to 'https://www.test.com:443'.

```
# url rewrite config - ex3
RewriteCond %{HTTP_HOST} ^www\.test\.com$
# if {Host} == "www.test.com"
RewriteCond %{SERVER_PORT} 80
# AND {Port} == "80"
RewriteRule .* https://www.test.com:443$0 [R]
# then > "https://www.test.com:443$0" ($0: request uri)
```

- **Example 4: Domain change (ignoring URI)**

The following is an example of a configuration that matches a URL pattern such as 'www.test.com/xxx.html' and changes it to 'www.test\_new.com/', ignoring the request URI.

```
# url rewrite config - ex4
RewriteCond %{HTTP_HOST} ^www\.test\.com$           # if {Host} == "www.test.com"
RewriteRule .* http://www.test_new.com [R]          # then > "http://www.test_new.com"
```

- **Example 5: Domain change (preserving URI)**

The following is an example configuration that matches a URL pattern like 'www.test.com/test.html' and changes it to 'www.test\_new.com/test.html', preserving the request URI.

```
# url rewrite config - ex5
RewriteCond %{HTTP_HOST} ^www\.test\.com$
# if {Host} == "www.test.com"
RewriteRule .* http://www.test_new.com$0 [R]
# then > "http://www.test_new.com$0" ($0: request uri)
```

- **Example 6: Forbidden error if no Referer header in POST requests**

The following is an example configuration that returns a "403 Forbidden" error when the request method is POST and the Referer header does not exist.

```
# url rewrite config - ex6
RewriteCond %{REQUEST_METHOD} POST                 # if {Method} == "POST"
RewriteCond %{HTTP_REFERER} =""                    # AND {Referer} == ""
RewriteRule . - [F]                                # then > 403 Forbidden Return
```

- **Example 7: URL redirection by subdomain**

The following is an example of a configuration that processes "www.test.com/aaa/xxx.html" if the request is "aaa.test.com/xxx.html" and processes "www.test.com/bbb/xxx.html" if the request is

"bbb.test.com/xxx.html".

```
# url rewrite config - ex8
RewriteCond %{HTTP_HOST} ^(aaa|bbb)\.test\.com
# if {Host} == ("aaa.test.com" OR "bbb.test.com")
RewriteRule .* /%1$0 [L]
# then "/xxx.html" > "/(aaa|bbb)/xxx.html" (%1: Opening parenthesis of RewriteCond, $0: request uri)
```

- **Example 8: Changing ports based on query string**

The following is an example of a configuration that rewrites

"www.test.com/test.do?query=value1" as "www.test.com:8080/test.do?query=value1" and

"www.test.com/test.do?query=value2&.." as "www.test.com:8080/test.do?query=value2&..".

```
# url rewrite config - ex9
RewriteCond %{QUERY_STRING} ^query=value1$ [OR]
# if {QueryString} == "query=value1"
RewriteCond %{QUERY_STRING} ^query=value2&
# OR {QueryString} == "query=value2&.."
RewriteRule .* http://www.test.com:8080$0 [R]
# then > "http://www.test.com:8080$0" ($0: request uri)
```

- **Example 9: Redirecting requests from specific subdomains to the www domain**

The following is an example a configuration that processes a request to "aaa.test.com/test.html" (not starting with 'www') by redirecting it to "www.test.com/aaa/test.html".

```
# url rewrite config - ex10
RewriteCond %{HTTP_HOST} !^www\.test\.com$
# if {Host} != www.test.com
RewriteCond %{HTTP_HOST} ^([a-zA-Z0-9]+)\.test\.com$
# AND {Host} == "(alphanumeric characters).test.com"
RewriteRule .* /%1/$0 [L]
# then > "(alphanumeric string)/$0" (%1: Opening parenthesis of RewriteCond, $0: request uri)
```

- **Example 10: Blocking CSS/JS requests without a Referer header**

The following is an example of a configuration that processes requests for CSS or JS files without a Referer header by returning a 403 Forbidden error.

```
# url rewrite config - ex11
RewriteCond %{HTTP_REFERER} !.          # if {HTTP_REFERER} == ""
RewriteRule \.(css|js)$ - [F]          # then "*.css|*.js" > 403 Forbidden Return
```

- **Example 11: Converting a specific pattern into a URL query parameter**

The followsig is an example configuration that matches a URL pattern "/user@somehost.com/"

and changes it to `/req_test.jsp?blogId=user@somehost.com`".

```
# url rewrite config - ex12 (REQUEST_URI)
RewriteCond %{REQUEST_URI} ^([a-zA-Z0-9_-]+)(@[a-zA-Z0-9_-]+)(com|net|co.kr))/?/?$
RewriteRule . /req_test.jsp?blogId=%1 [PT,L]
```

- **Example 12: Redirection based on specific host and URI pattern**

The following is an example of a configuration that redirects requests when the HTTP request header's host ends with `"tmaxsoft.com"` and the URL starts with `"/redirect/"` to `"http://www.tmaxsoft.com/redirect.html"`.

```
# url rewrite config - ex13 (HTTP_HOST)
RewriteCond %{HTTP_HOST} tmaxsoft.com$
RewriteCond %{REQUEST_URI} /redirect/.*$
RewriteRule . http://www.tmaxsoft.com/redirect.html [R,L]
```

## 6. WebtoB Admin API

WebtoB provides management features through the Admin API using Rest API. A port can be specified in the /server/admin settings, and API communication is performed through that port.

The admin settings in config are configured as follows. In the following example settings, Admin API communication is possible on port 9090.

```
"server":{
  "admin": {
    "port": 9090
  }
}
```



If `/node/process_count` is greater than 1, the Engine's Admin port is calculated by adding the Engine index to the specified value.

For example, when `process_count` is 3 and `port` is 9090, the Admin port for the Engine with index 2 is 9092.

The following describes the APIs provided by the Admin API.

| API                             | Description                                                                  |
|---------------------------------|------------------------------------------------------------------------------|
| <a href="#">/config</a>         | Displays environment configuration settings.                                 |
| <a href="#">/updateconfig</a>   | Reloads the config file to dynamically apply settings.                       |
| <a href="#">/webtobinfo</a>     | Displays the WebtoB system information.                                      |
| <a href="#">/patchinfo</a>      | Retrieves information about patches applied after the release.               |
| <a href="#">/cliinfo</a>        | Checks the connected web browser.                                            |
| <a href="#">/connectioninfo</a> | Displays connection information.                                             |
| <a href="#">/svginfo</a>        | Checks the destination information.                                          |
| <a href="#">/statinfo</a>       | Retrieves statistics about the state of destinations, threads, and services. |
| <a href="#">/restat</a>         | Resets statistics for the destination.                                       |
| <a href="#">/shutdown</a>       | Shuts down WebtoB.                                                           |
| <a href="#">/cachelist</a>      | Displays information about responses stored in the HTTP response cache.      |
| <a href="#">/cacherefresh</a>   | Deletes responses stored in the HTTP response cache.                         |
| <a href="#">/certrefresh</a>    | Refreshes the certificate information.                                       |

The main features of WebtoB Admin API are as follows:

- All APIs communicate using the POST method.
- For APIs operating on HTH, if there is no response from HTH for a certain period of time, it will be displayed as 'busy\_hth' after a timeout. For example, if HTH-1 and HTH-2 do not respond within the timeout period for the /cliinfo API, the information will be displayed as 'busy\_hth' as shown in the following example.

```
{
  "result": {
    "HTH-0": [
      {
        ...
      }
    ]
  },
  "busy_hth": [
    "HTH-1",
    "HTH-2"
  ]
}
```

- If the API operates correctly, it returns HTTP response code 200, otherwise it returns an appropriate response code depending on the situation.

The following describes common error response codes.

| Response Code | Description                                                                                          |
|---------------|------------------------------------------------------------------------------------------------------|
| 400           | Failed to validate JSON schema for HTTP body.                                                        |
| 401           | User authentication failed.                                                                          |
| 403           | No permission to perform actions on the requested API.                                               |
| 404           | Requested API not found.                                                                             |
| 405           | Invalid method used for the request.                                                                 |
| 500           | An error occurred during API operation. (The specific error and message differ depending on the API) |

- If the response code is not 200, it is commonly structured as follows:

```
{
  "error_code": string,
  "error_message": string
}
```

For example, a response code of 404 may appear as follows:

```
{
  "error_code": "APIERR_COMMON_0003",
  "error_message": "No such API"
}
```

```
}
```

## 6.1. Environment Information

### 6.1.1. /config

Retrieves the environment information of the currently running system. This process allows you to check not only the values specified in the environment settings file but also the values applied by default.

- Usage

```
{
  "path": string
}
```

| Option | Description                                                                                                                                                                                       |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Path   | Specifies the JSON path to the environment configuration file. The path must always start with '/'.<br>If the path is omitted or set to '/', all currently configured settings will be retrieved. |

- Example

The following is an example of checking the NODE section environment settings with the "/config" Admin API. For detailed information of the configuration items in the NODE section, refer to [Configuration Items](#).

```
[Request]
{
  "path": "/node"
}

[Response]
{
  "result": {
    "cache_entry": 128,
    "cache_key": "HOST_URI",
    "cache_max_file_size": 8192,
    "connection_pool_size": 8192,
    "graceful_shutdown_timeout": 5,
    "hth_count": 1,
    "hth_schedule": "RR",
    "limit_request_body_size": 0,
    "limit_request_header_field_count": 100,
    "limit_request_header_field_size": 8190,
    "limit_request_line_size": 8190,
    "listen_backlog": 4096,
```

```

        "max_cache_memory_size": 100,
        "name": "webtob-server",
        "system_filters": [],
        "worker_threads": 1
    }
}

```

### 6.1.2. /updateconfig

Reloads the configuration file and apply the settings dynamically. Only settings specified with the '|dynamic|' key in the schema file are applied.

- Usage

This API does not require any separate parameters.

- Example

The following is an example of updating the settings using the "/updateconfig" Admin API after modifying the config file.

```

[Request]
{ }

[Response]
{ }

```

### 6.1.3. /webtobinfo

Displays general information about the currently running WebtoB.

- Usage

This API does not require any separate parameters.

- Example

The following is an example of checking information about the currently running WebtoB using the "/webtobinfo" Admin API.

```

[Request]
{}

[Response]
{
    "result": {
        "license_type": "CLOUD Enterprise edition",
        "expiration_date": "2099/12/31",
        "hth_count": 1,
        "worker_thread_count": 1
    }
}

```

```
}  
}
```

### 6.1.4. /patchinfo

Retrieves patch information applied after the release.

- Usage

```
{  
  "patch_version": string  
}
```

| Option        | Description                                                                                                   |
|---------------|---------------------------------------------------------------------------------------------------------------|
| patch_version | Path version name or version number.<br><br>If omitted, all changes made after the release will be retrieved. |

- Example

The following is an example of checking patch information for version B18 using the "/patchinfo" Admin API.

```
[Request]  
{  
  "patch_version": "6.0_B18"  
}  
  
[Response]  
{  
  "result": [  
    "WebtoB 6 Linux B19 epoll 2025.04.07",  
    {  
      "* 6_B18 (2025.03.31)": [  
        "Added the PatchInfo command",  
        "Changed the location of wscfl (bin->lib)",  
        "Fixed an issue where METHOD was not migrated when the ACCESS section is missing  
in the .m file",  
        "Fixed an issue where the license was read from an iccorect path",  
        "Modified wsreboot.sh to run outside the bin directory"  
      ]  
    }  
  ]  
}
```

## 6.2. Operational Status Information

### 6.2.1. /cliinfo

Displays the environment settings, such as the current status, IP address, number of processed requests, of the currently connected client (usually a web browser).

- Usage

```
{
  "http_server": string,
  "virtual_host": string,
  "hth_number": integer
}
```

| Option       | Description                                                                                                                                                                                                |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| None         | Retrieves information about all clients connected to the server.                                                                                                                                           |
| http_server  | Retrieves information about clients connected to the specified HTTP server.                                                                                                                                |
| virtual_host | Retrieves information about the clients connected to the specified virtual host. This option can only be set if http_server is set, and the value must be a virtual host name included in the http_server. |
| hth_number   | Retrieves information about clients connected to the specified HTH. If not specified, information for all HTHs will be retrieved.                                                                          |

- Example

The following is an example of checking the client information connected to the vhost1 virtual host in HTH 3 using the "/cliinfo" Admin API.

```
[Request]
{
  "http_server": "http1",
  "virtual_host": "vhost1",
  "hth_number": 3
}

[Response]
{
  "result": {
    "HTH-3": [
      {
        "no": 1,
        "idle": 24,
        "local_ipaddr:port": "192.168.15.167:8080",
        "remote_ipaddr:port": "192.168.15.168:48006",
        "request_count": 0,
        "response_count": 0,
        "ssl": false,
```

```

    "status": "READY",
    "http_server": "http1",
    "virtual_host": "vhost1"
  },
  {
    "no": 0,
    "idle": 24,
    "local_ipaddr:port": "192.168.15.167:8080",
    "remote_ipaddr:port": "192.168.15.168:48005",
    "request_count": 2,
    "response_count": 2,
    "ssl": false,
    "status": "READY",
    "http_server": "http1",
    "virtual_host": "vhost1"
  }
]
}
}

```

The following describes each output item.

| Item                                     | Description                                                                                                                                                                                                                                                                                                                                                                                                     |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| no                                       | The connection number managed internally by WebtoB.                                                                                                                                                                                                                                                                                                                                                             |
| idle                                     | Client idle time with no data exchanges.                                                                                                                                                                                                                                                                                                                                                                        |
| local_ipaddr:port,<br>remote_ipaddr:port | IP address and port number of the server and client.                                                                                                                                                                                                                                                                                                                                                            |
| request_count                            | The number of requests sent by the client.                                                                                                                                                                                                                                                                                                                                                                      |
| response_count                           | The number of responses sent to the client.                                                                                                                                                                                                                                                                                                                                                                     |
| ssl                                      | Specifies whether the client is using an SSL connection.                                                                                                                                                                                                                                                                                                                                                        |
| status                                   | Client state within the server. <ul style="list-style-type: none"> <li>◦ CONNECTING: Trying to connect.</li> <li>◦ CONNECTED: Connection established.</li> <li>◦ READY: Receiving requests from the client.</li> <li>◦ RUNNING: A client request is being processed on the server.</li> <li>◦ DISCONNECTING: Attempting to terminate the connection.</li> <li>◦ DISCONNECTED: Connection terminated.</li> </ul> |
| http_server                              | The HTTP server that the client connected to.                                                                                                                                                                                                                                                                                                                                                                   |
| virtual_host                             | The last virtual host accessed by the client.                                                                                                                                                                                                                                                                                                                                                                   |

## 6.2.2. /connectioninfo

Displays the environment settings, such as the current status, IP address, number of processed

requests, of the currently connected client (usually a web browser) and server.

- Usage

```
{
  "hth_number": integer
}
```

| Option     | Description                                                                                                                              |
|------------|------------------------------------------------------------------------------------------------------------------------------------------|
| None       | Retrieves information about all clients connected to WebtoB.                                                                             |
| hth_number | Retrieves information about clients and servers connected to the specified HTH. If not specified, information for all HTHs is retrieved. |

- Example

The following is an example of checking the client and server information connected to HTH 3 using the "/connectioninfo" Admin API.

```
[Request]
{
  "hth_number": 3
}

[Response]
{
  "result": {
    "HTH-3": [
      {
        "acceptor": "WJP-0",
        "request_count": 1,
        "response_count": 1,
        "counterpart": "",
        "idle": 3,
        "local_ipaddr:port": "127.0.0.1:9999",
        "no": 0,
        "remote_ipaddr:port": "127.0.0.1:44866",
        "remote_type": "JSV",
        "ssl": false,
        "status": "READY",
        "virtual_host": ""
      },
      {
        "acceptor": "http1",
        "request_count": 2,
        "response_count": 2,
        "counterpart": "",
        "idle": 2,
        "local_ipaddr:port": "192.168.15.167:8080",
        "no": 10,
        "remote_ipaddr:port": "192.168.15.168:64223",
        "remote_type": "CLIENT",
        "ssl": false,
        "status": "READY",
```

```

    "virtual_host": ""
  },
  {
    "acceptor": "",
    "request_count": 1,
    "response_count": 1,
    "counterpart": "",
    "idle": 2,
    "local_ipaddr:port": "192.168.15.167:60504",
    "no": 11,
    "remote_ipaddr:port": "192.168.15.124:8080",
    "remote_type": "RPROXY",
    "ssl": false,
    "status": "READY",
    "virtual_host": ""
  }
]
}

```

The following describes each output item.

| Item                                     | Description                                                                                                                                                                                                                                                                                                                                                                                                     |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| no                                       | The connection number managed internally by WebtoB.                                                                                                                                                                                                                                                                                                                                                             |
| idle                                     | Idle time with no data exchanges.                                                                                                                                                                                                                                                                                                                                                                               |
| local_ipaddr:port,<br>remote_ipaddr:port | IP address and port number of the server and client.                                                                                                                                                                                                                                                                                                                                                            |
| request_count                            | The number of requests sent through this connection.                                                                                                                                                                                                                                                                                                                                                            |
| response_count                           | The number of responses sent through this connection.                                                                                                                                                                                                                                                                                                                                                           |
| ssl                                      | Specifies whether the connection is established using SSL.                                                                                                                                                                                                                                                                                                                                                      |
| status                                   | Client state within the server. <ul style="list-style-type: none"> <li>◦ CONNECTING: Trying to connect.</li> <li>◦ CONNECTED: Connection established.</li> <li>◦ READY: Receiving requests from the client.</li> <li>◦ RUNNING: A client request is being processed on the server.</li> <li>◦ DISCONNECTING: Attempting to terminate the connection.</li> <li>◦ DISCONNECTED: Connection terminated.</li> </ul> |
| acceptor                                 | The server that this connection is connected to.                                                                                                                                                                                                                                                                                                                                                                |
| virtual_host                             | The last virtual host accessed by this connection. (Applies only to clients)                                                                                                                                                                                                                                                                                                                                    |
| counterpart                              | The number of the counterpart to this connection. If the current connection is a client, the counterpart refers to a server connection, and vice versa. For queued WJP connection, it may be displayed as -1.                                                                                                                                                                                                   |

## 6.2.3. /svginfo

Displays information about each destination that is currently in operation.

- Usage

```
{
  "htmls": boolean,
  "jeus" : array(string),
  "rproxy" : array(string)
}
```

| Option | Description                                                                                                                      |
|--------|----------------------------------------------------------------------------------------------------------------------------------|
| None   | Displays information about all destinations.                                                                                     |
| html   | Sets whether to display HTMLS destination information.                                                                           |
| jeus   | Specifies the JEUS destination name. Including "*" displays information for all JEUS destinations.                               |
| rproxy | Specifies the reverse proxy group destination name. Including "*" displays information for all reverse proxy group destinations. |

- Example

The following is an example of checking information for all JEUS destinations and the reverse proxy group destination named "rproxy1" using the "/svginfo" Admin API.

```
[Request]
{
  "jeus" : [*],
  "rproxy" : ["rproxy1"]
}

[Response]
"result": [
  {
    "HTH-0": {
      "jeus": [
        {
          "aqcnt": 0,
          "count": 0,
          "cqcnt": 0,
          "qpcent": 0,
          "reqs": 0,
          "rscent": 0,
          "status": "NRDY",
          "svgname": "MyGroup2"
        },
        {
          "aqcnt": 0,
          "count": 0,
          "cqcnt": 0,
          "qpcent": 0,
```

```

        "reqs": 0,
        "rsent": 0,
        "status": "NRDY",
        "svgname": "MyGroup1"
    }
],
"rproxy": [
    {
        "aqcnt": 0,
        "count": 0,
        "cqcnt": 0,
        "qpcent": 0,
        "reqs": 0,
        "rsent": 0,
        "status": "RDY",
        "svgname": "rproxy1"
    }
]
}
}
]

```

The following describes each output item.

| Item    | Description                                                                                                                                                                                                                                                                                                                                                                                               |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| aqcnt   | Cumulative number of queued requests. (Cumulative cqcnt value)                                                                                                                                                                                                                                                                                                                                            |
| count   | Number of processed requests.                                                                                                                                                                                                                                                                                                                                                                             |
| cqcnt   | Number of requests in the queue.                                                                                                                                                                                                                                                                                                                                                                          |
| qpcent  | Number of queued requests that have been removed from the queue due to request timeout, qp command, etc.                                                                                                                                                                                                                                                                                                  |
| reqs    | Number of requests sent to the destination.                                                                                                                                                                                                                                                                                                                                                               |
| rsent   | Number of server restarts due to abnormal termination of the destination.                                                                                                                                                                                                                                                                                                                                 |
| status  | Client state within the destination. <ul style="list-style-type: none"> <li>◦ RDY: Request can be processed by the destination. There are destination processes connected to WebtoB.</li> <li>◦ NRDY: Request cannot be processed. There are no destination processes connected to WebtoB.</li> <li>◦ BLK: The destination is suspended by administrator command. Request cannot be processed.</li> </ul> |
| svgname | Destination name specified in configuration.                                                                                                                                                                                                                                                                                                                                                              |

## 6.2.4. /statinfo

Displays actual system operational status and retrieves information about the destination processes

and services statistics currently in operation.

You can check dynamic information for each destination, such as the number of services processed and the number of connections.

- Usage

```
{
  "hth_number" : integer,
  "target":{
    "jeus": array(string),
    "rproxy" : array(string),
    "htmls": boolean
  }
}
```

| Option        | Description                                                                                                                                                             |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| hth_number    | Specifies the HTH number for which to check statistics.                                                                                                                 |
| target/jeus   | Specifies the JEUS destination for which statistical information is displayed. If "*" is included, information for all JEUS destinations is displayed.                  |
| target/rproxy | Specifies the reverse proxy destination for which statistical information is displayed. If '*' is included, information for all reverse proxy destination is displayed. |
| target/htmls  | Specifies whether to display HTMLS statistical information.                                                                                                             |



At least one item must be specified under 'target'.

- Example

The following is an example of checking information for the JEUS destination named "MyGroup1", the destination within the reverse proxy group named "rproxy1", and the HTMLS destination using the "/statinfo" Admin API.

```
[Request]
{
  "hth_number" : 1,
  "target":{
    "jeus": ["MyGroup1"],
    "rproxy" : ["rproxy1"],
    "htmls": true
  }
}

[Response]
{
  "result": {
    "HTH-0": {
      "jeus": [
```

```

    {
      "average_processed_time": 0.0,
      "connections": 50,
      "request_count": 0,
      "response_count": 0,
      "server": "amV1c19kb21haW4vc2VydMvYMQ==",
      "server_group": "MyGroup1",
      "sticky_routed_count": 0
    }
  ],
  "rproxy": [
    {
      "average_processed_time": 0.0,
      "connections": 0,
      "request_count": 0,
      "response_count": 0,
      "server": "192.168.15.167:8090",
      "server_group": "rproxy1",
      "sticky_routed_count": 0
    },
    {
      "average_processed_time": 0.0,
      "connections": 0,
      "request_count": 0,
      "response_count": 0,
      "server": "192.168.15.167:8088",
      "server_group": "rproxy1",
      "sticky_routed_count": 0
    }
  ],
  "htmls": {
    "average_processed_time": 0.000309519,
    "queue_count": 0,
    "request_count": 5,
    "response_count": 5,
    "server": "HTMLS"
  }
}

```

The following describes each output item.

| Item                   | Description                                              |
|------------------------|----------------------------------------------------------|
| average_processed_time | The average processing time per request. (Unit: seconds) |
| connections            | The number of connections linked to the destination.     |
| request_count          | The number of requests sent to the destination.          |
| response_count         | The number of requests processed by the destination.     |

| Item                | Description                                                                                                                                                                                     |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server              | Destination identification information <ul style="list-style-type: none"> <li>◦ jeus: jengineid</li> <li>◦ Reverse proxy: Target destination address</li> <li>◦ HTMLS: HTMLS (fixed)</li> </ul> |
| server_group        | The destination name                                                                                                                                                                            |
| sticky_routed_count | The number of times the request was routed to the destination based on the Sticky ID received from the client.                                                                                  |

### 6.2.5. /restat

Resets statistical information for the destination. Use this command to reset the information for the associated services or destinations.

- Usage

```
{
  "hth_number" : integer,
  "target":{
    "jeus": array(string),
    "rproxy" : array(string),
    "htmls": boolean
  }
}
```

| Option        | Description                                                                                                      |
|---------------|------------------------------------------------------------------------------------------------------------------|
| None          | Resets all HTHs.                                                                                                 |
| hth_number    | Specifies the HTH number to reset. If not set, all HTHs are reset.                                               |
| target        | Specifies the target to reset. At least one target must be specified.                                            |
| target/jeus   | Specifies the JEUS destinations to reset. If '*' is included, all JEUS destinations are reset.                   |
| target/rproxy | Specifies the reverse proxy destinations to reset. If '*' is included, all reverse proxy destinations are reset. |
| target/htmls  | Specifies whether to reset HTMLS statistics.                                                                     |

- Example

The following is an example of resetting statistics for the JEUS destination named "MyGroup1".

```
[Request]
{
  "target": {
```

```

    "jeus": ["MyGroup1"]
  }
}

[Response]
"HTH-0": {
  "jeus": [
    "MyGroup1"
  ]
}

```



Returns the names of the targets that were successfully reset for each HTH.

## 6.3. Process Shutdown

### 6.3.1. /shutdown

Shuts down the running WebtoB.

- Usage

```

{
  "timeout": integer
}

```

| Option    | Description                                                                                                                                                                                                                             |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| "timeout" | Specifies the timeout for the graceful shutdown process in seconds. If WebtoB does not shut down within the specified time, it will be forcibly terminated.<br>If omitted, the value set in /node/graceful_shutdown_timeout is applied. |

- Example

The following is an example of calling the "/shutdown" Admin API to forcibly terminate WebtoB if it hasn't shut down after attempting a graceful shutdown for 30 seconds.

```

[Request]
{
  "timeout": 30
}

[Response]
{
  "result": {}
}

```



The response is returned immediately after WebtoB receives the command. However, the WebtoB process may still be running even after the response is received.

## 6.4. Cache Information Management

### 6.4.1. /cachelist

Displays the response information currently stored in WebtoB's HTTP response cache.

- Usage

```
{
  "hth_number": integer
}
```

| Option       | Description                                          |
|--------------|------------------------------------------------------|
| None         | Displays all cache information.                      |
| "hth_number" | Specifies the HTH number to query cache information. |

- Example

The following is an example of checking cache information for HTH 0 using the "/cachelist" Admin API. The rest of the request path is used only for internal server debugging.

```
[Request]
{
  "hth_number": 0
}

[Response]
"result": {
  "HTH-0": {
    "cache_map": [
      {
        "expire_time": "2023-10-25 16:35:18",
        "key": "192.168.15.167:8080/index.html",
        "size": 4312
      },
      {
        "expire_time": "2023-10-25 16:34:52",
        "key": "192.168.15.167:8080/favicon.ico",
        "size": 4507
      },
      {
        "expire_time": "2023-10-25 16:34:52",
```

```

        "key": "192.168.15.167:8080/4.html",
        "size": 219
    },
    ],
    "cache_map_size": 3,
    "memory_usage": 9038
}

```

The following describes each output item.

| Item                  | Description                                                               |
|-----------------------|---------------------------------------------------------------------------|
| cache_map             | Returns a key (URI) and related information managed in the cache.         |
| cache_map/expire_time | The expiration time of the item.                                          |
| cache_map/key         | The URI of the item.                                                      |
| cache_map/size        | The total size of the cache item.                                         |
| cache_map_size        | The number of cache items currently managed in the cache.                 |
| memory_usage          | The total amount of memory used by the cache item content. (Unit: bytes). |

## 6.4.2. /cacherefresh

Removes the response data from the WebtoB HTTP response cache.

- Usage

```

{
  "url": string,
  "ext": string
}

```

| Option | Description                                         |
|--------|-----------------------------------------------------|
| None   | Clears the entire cache.                            |
| url    | Specifies the URL of the cache to be removed.       |
| ext    | Specifies the extension of the cache to be removed. |



Only one of url or ext can be specified.

The following is an example of removing the response for "test.domain.com/test.html" from the cache using the "/cacherefresh" Admin API.

[Request]

```
{
  "url": "test.domain.com/test.html"
}

[Response]
"result": {
  "HTH-0": {
    "removed_count": 1
  }
}
```

The following describes each output item.

| Item          | Description                                  |
|---------------|----------------------------------------------|
| removed_count | The number of cache items that were removed. |

## 6.5. Certificate Management

### 6.5.1. /certrefresh

Refreshes WebtoB's certificate information.

- Usage

This API does not require any separate parameters.

- Example

The following is an example of refreshing certificate information.

```
[Request]
{ }

[Response]
{ }
```

## 7. WebtoB Console Tool

WebtoB provides the following tools to manage engine processes and server processes.

- **Admin Tools**

| Console Tool            | Description                                                                                                            |
|-------------------------|------------------------------------------------------------------------------------------------------------------------|
| <a href="#">wsadmin</a> | A tool used for the overall WebtoB system management, supporting system information retrieval and administrator tasks. |

- **Other Tools**

| Console Tool                    | Description                                                 |
|---------------------------------|-------------------------------------------------------------|
| <a href="#">configMigrator</a>  | Migrates WebtoB 5 configuration files to WebtoB 6.          |
| <a href="#">configValidator</a> | Performs schema validation on WebtoB environment files.     |
| <a href="#">mkpwd</a>           | Creates a file to store passwords for SSL certificate keys. |

### 7.1. wsadmin

wsadmin provides a text-based administration environment. It always waits at a prompt and executes user-entered commands.

- Execution

To run the wsadmin tool, use the **wsadmin** command.

```
$ wsadmin
```

If the wsadmin tool runs successfully, the following message is displayed at the prompt:

```
$$1 [wsadmin]>>
```

- Exit

To exit the wsadmin tool, use the **exit** command.

```
$$3 [wsadmin]>> exit
```

The following describes the commands provided by wsadmin.

| Command                         | Abbreviation | Description                                                                  |
|---------------------------------|--------------|------------------------------------------------------------------------------|
| <a href="#">help</a>            |              | Displays help for commands available in the wsadmin tool.                    |
| <a href="#">client-info</a>     | (cli)        | Displays client connection information for WebtoB.                           |
| <a href="#">connect</a>         |              | Connects to a specific WebtoB Admin server.                                  |
| <a href="#">stat-info</a>       | (st)         | Displays statistics about destination and service status.                    |
| <a href="#">config</a>          | (cfg)        | Displays environment settings information.                                   |
| <a href="#">update-config</a>   |              | Reloads the configuration file to dynamically apply settings.                |
| <a href="#">cache-list</a>      | (cachelist)  | Displays information about the HTTP response cache.                          |
| <a href="#">cache-refresh</a>   | (cr)         | Deletes responses stored in the HTTP response cache.                         |
| <a href="#">cert-refresh</a>    | (cr)         | Refreshes WebtoB's certificate information.                                  |
| <a href="#">svg-info</a>        | (and)        | Displays information about the destination that handles the client requests. |
| <a href="#">connection-info</a> | (There)      | Displays connection information between WebtoB's clients and destinations.   |
| <a href="#">shutdown</a>        | (down)       | Shuts down the WebtoB.                                                       |
| <a href="#">exit</a>            |              | Exits the wsadmin tool.                                                      |
| <a href="#">patch-info</a>      |              | Retrieves patch information applied after the release.                       |
| <a href="#">webtob-info</a>     |              | Displays the WebtoB system information.                                      |
| <a href="#">license-info</a>    |              | Displays the specified license information.                                  |
| <a href="#">system-info</a>     |              | Displays information about the current system.                               |

Each command provides the following common options:

| Option                                          | Description                                                                                  |
|-------------------------------------------------|----------------------------------------------------------------------------------------------|
| <code>[-i &lt;repeat interval&gt;]</code>       | Specifies the repetition interval in seconds of the command. (Default: 1 second)             |
| <code>[-r &lt;number of repetitions&gt;]</code> | Specifies the number of repetitions. (Default: 0)<br>If set to 0, it will repeat infinitely. |

The following is an example of executing the cli command 2 times with a 2-second interval.

```
wsadmin]>> cli -i 2 -r 2
Command will be executed repeatedly. Count=2. Interval=2(sec).
* Repeat count = 1
```

```
-----
----
| Client Info : HTH-0
|
-----
----
| No | Status | Request Count | Response Count | Idle | Local Address | Remote Address |
Ssl |
```

```

-----
| 12 | READY |      1      |      1      | 22 | 192.168.15.120:8080 | 192.168.15.120:53844 |
No |
-----

* Repeat count = 2

-----
| Client Info : HTH-0
|
-----

-----
| No | Status | Request Count | Response Count | Idle | Local Address | Remote Address |
Ssl |
-----

| 12 | READY |      1      |      1      | 24 | 192.168.15.120:8080 | 192.168.15.120:53844 |
No |
-----

Repeat done.

```

When `wsadmin` is executed with commands enclosed in double quotation marks ( " ), the results of each command are output sequentially without waiting at a prompt.

The following example shows how the results of the `cli` and `st` commands are output directly to the console.

```

$ wsadmin "cli" "st"
  Environment variables for WebtoB:
    WEBTOB_HOME_PATH = /home/tmax/webtob6/

Welcome to WebtoB Demo system.
Enterprise edition License in used.
Schema file path: /home/tmax/webtob6/schema/webtob-config-enterprise.schema.json

Start config validation
1. Opened schema file
2. Opened config file
3. Config parsing finished
4. Pre-validation process finished
5. Validation using json-schema finished
6. Post-validation process finished
Config file "/home/tmax/webtob6/config/webtob-config.json" is validated with schema file
"/home/tmax/webtob6/schema/webtob-config-enterprise.schema.json"
Success to load config files : webtob-config.json
Validation success
  Current WebtoB Configuration:
    Number of client handlers (HTH) = 1
    Number of worker threads per handler (HTH) = 1
    Number of connection pool size per handler (HTH) = 100

```

Today is 2025-05-28  
License expires on 2099-12-31 (27245 days remaining)  
Connection established successfully with localhost:9090.

-----  
| Client Info : HTH-0 |

-----  
| No | Status | Request Count | Response Count | Idle | Local Address | Remote Address | Ssl |

-----  
Prints all information. Use options to print specific items ( -jeus, -rproxy, -htmls )

-----  
| HTH-0 : JEUS statistic

-----  
| Server | Server Group | Connections | Average Processed Time | Request Count |  
| Response Count | Sticky Routed Count |

-----  
| ZG9tYWluMS9hZG1pb1NlcjZlcg== | MyGroup1 | 10 | 0.000000 | 0  
| 0 | 0 |

-----  
| HTH-0 : Reverse Proxy statistic

-----  
| Server | Server Group | Connections | Average Processed Time | Request Count | Response  
Count | Sticky Routed Count |

-----  
| tmaxsoft:80 | rproxyg1 | 0 | 0.000000 | 0 | 0  
| 0 |

-----  
| HTH-0 : HTMLS statistic

-----  
| Server | Queue Count | Average Processed Time | Request Count | Response Count |

-----  
| HTMLS | 0 | 0.000000 | 0 | 0 |

### 7.1.1. help

Displays the help for commands available in the wsadmin tool.

- Usage

```
> help [command]
```

| Option    | Description                                                  |
|-----------|--------------------------------------------------------------|
| [Command] | Specifies the command for which to output usage information. |

- Example

#### No Options Used

```
[wsadmin]>> help
```

|                                                           |                   |                                                     |  |
|-----------------------------------------------------------|-------------------|-----------------------------------------------------|--|
| -----                                                     |                   |                                                     |  |
| Help                                                      |                   |                                                     |  |
| -----                                                     |                   |                                                     |  |
| Command Name                                              | Command Alias     | Command Descriptor                                  |  |
| -----                                                     |                   |                                                     |  |
| client-info                                               | cliinfo, cli      | show client info                                    |  |
| connect                                                   |                   | connect to the WebtoB admin server                  |  |
| stat-info                                                 | stat, st          | show statistics info                                |  |
|                                                           |                   | JEUS, Reverse proxy server, HTMLS statistic         |  |
| restat                                                    |                   | clear target statistics info                        |  |
| config                                                    | cfg               | show config info                                    |  |
| update-config                                             |                   | update config info                                  |  |
| cache-list                                                | cachelist         | show cache info                                     |  |
| cache-refresh                                             | cr, cacherefresh  | removes responses stored in the HTTP response cache |  |
| svg-info                                                  | svginfo, sgi, si  | show server group info                              |  |
| connection-info                                           | coninfo, conl, ci | show connection info                                |  |
| shutdown                                                  | down              | shutdown WebtoB                                     |  |
| -----                                                     |                   |                                                     |  |
| You can use the help command for each individual command. |                   |                                                     |  |
| -----                                                     |                   |                                                     |  |

#### Options Used

```
[wsadmin]>> help st
```

|                              |            |         |                              |
|------------------------------|------------|---------|------------------------------|
| -----                        |            |         |                              |
| Help : stat-info( stat, st ) |            |         |                              |
|                              |            |         |                              |
| -----                        |            |         |                              |
| OptionName                   | IsRequired | HasArgs | Description                  |
|                              |            |         |                              |
| -----                        |            |         |                              |
| hth_number                   | No         | Yes     | target hth number            |
|                              |            |         |                              |
| jeus                         | No         | Yes     | JEUS statistic info          |
|                              |            |         |                              |
| rproxy                       | No         | Yes     | Reverse proxy statistic info |
|                              |            |         |                              |
| htmls                        | No         | No      | HTMLS statistic info         |
|                              |            |         |                              |
| ---                          | ---        | ---     | ---                          |
|                              |            |         |                              |

```

| i      |      No      |      Yes      | Default: 1 (second) if a repeat is needed. Interval
specifies the time delay between repeats.      |
| r      |      No      |      Yes      | Default: 0 (unlimited) if a repeat is needed. To limit the
repeats, set a value greater than 0. |
-----
| Show statistics info.
|
| JEUS, Reverser Proxy, HTMLS statistic.
|
-----
-----

```

## 7.1.2. client-info (cli, cliinfo)

Displays the environment settings, such as the current status, IP address, number of processed requests, of the currently connected client (usually a web browser).

- Usage

```
> cli [-vhost <VHOST name>][-hth <HTH number>]
```

| Option                | Description                                                              |
|-----------------------|--------------------------------------------------------------------------|
| [-vhost <VHOST name>] | Specifies the VHOST name for which client information will be displayed. |
| [-hth <HTH number>]   | Specifies the HTH number for which client information will be displayed. |

- Example

Using the cli without options displays the following result. Each entry represents one client.

```

[wsadmin]>> cli
-----
| Client Info : HTH-0
|
-----
-----
| No | Status | Request Count | Response Count | Idle | Local Address | Remote Address
| Ssl |
-----
| 5 | READY | 0 | 0 | 4 | 192.168.15.120:8080 |
192.168.15.120:38572 | No |
-----
-----

```

The following describes each output item.

| Item                          | Description                                                                                                                                                                                                  |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| No                            | The connection number managed internally by WebtoB.                                                                                                                                                          |
| Status                        | Client status for the destination <ul style="list-style-type: none"> <li>◦ READY: Receiving requests from the client.</li> <li>◦ RUNNING: A client request is being processed at the destination.</li> </ul> |
| Request Count                 | The number of requests sent by the client.                                                                                                                                                                   |
| Response Count                | The number of responses received by that client.                                                                                                                                                             |
| Idle                          | Client idle time with no data exchanges.                                                                                                                                                                     |
| Local Address, Remote Address | IP addresses and port numbers of the server and client                                                                                                                                                       |
| Ssl                           | Specifies whether the client is using an SSL connection.                                                                                                                                                     |

### 7.1.3. connect

This command is used to connect to the WebtoB Admin server from wsadmin.

- Usage

```
> connect [-ip <Admin server IP>][-port <Admin server port>]
```

| Option                      | Description                                                         |
|-----------------------------|---------------------------------------------------------------------|
| [-ip <Admin server IP>]     | Specifies the IP address of the WebtoB Admin server to connect to.  |
| [-port <Admin server port>] | Specifies the port number of the WebtoB Admin server to connect to. |

- Example

```
[wsadmin]>> connect -ip localhost -port 9090
Connection established successfully with localhost:9090.
```

### 7.1.4. patch-info (patchinfo)

Retrieves information about patches applied after the release.

- Usage

```
> patch-info [-version <version information>]
```

| Option                           | Description                                                                                                   |
|----------------------------------|---------------------------------------------------------------------------------------------------------------|
| [-version <version information>] | Path version name or version number.<br><br>If omitted, all changes made after the release will be retrieved. |

- Example

```
[wsadmin]>> patch-info -version 6.0_B18
```

```
-----
| Patch info                                     |
|-----|
| WebtoB 6 Linux B19 epoll 2025.04.07         |
|-----|
| * 6_B18 (2025.03.31)                         |
|   - Added the PatchInfo command              |
|   - Fixed an issue                          |
|-----|
|-----|
```

### 7.1.5. webtob-info (webtobinfo, wi)

Displays general information about the currently running WebtoB.

- Usage

This command does not require any separate options.

- Example

```
[wsadmin]>> webtob-info
```

```
-----
| WebtoB Version: WebtoB 6 Linux B19 epoll 2025.04.07 |
|-----|
```

```
-----
| License Type      | Expiration Date | HTH Count | Worker Thread Count | Connection Pool |
| Size |           |              |          |                   |               |
|-----|
| Cloud Enterprise edition | 0000/00/00      | 1          | 8              | 2              |
|-----|
```

## 7.1.6. license-info (licenseinfo, li)

Displays information about a specific license. This command can be executed even when WebtoB is not running.

- Usage

```
> license-info [-file <file path/file name>]
```

| Option                        | Description                                                                                                                                                                                                                                                                |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| [-file <file path/file name>] | Specifies the file path and file name of the license.<br><br>If the path does not start with '/', it is resolved relative to the the current directory where wsadmin was executed. If the path ends with '/', the file name <b>license.dat</b> is automatically specified. |

- Example

```
[wsadmin]>> license-info -file ../license/
-----
| License info (file: license.dat)                                     |
-----
| Sequence Number | Issue Date | Type | Edition | Expiration Date | Policy |
-----
| AAA-0000-000-0000 | 0000/00/00 | Cloud | Enterprise | 0000/00/00 | Unlimited |
-----
```

## 7.1.7. system-info (systeminfo)

Displays general information about the current system, including the host name. This command can be executed even when WebtoB is not running.

- Usage

This command does not require any separate options.

- Example

```
[wsadmin]>> systeminfo
-----
| System info                                                         |
-----
| Host Name | Host ID | Number of CPUs | Open Files |
-----
| webtob0  | 000A0000 | 12             | 1024/1048576 |
-----
-----
```

```
-----  
| Kernel and Hardware Details  
|  
-----
```

```
-----  
| Kernel Name | Kernel Release | Machine Hardware Name | Processor Type | Hardware Platform |  
Operating System |  
-----
```

```
-----  
| Linux | 6.8.0-52-generic | x86_64 | x86_64 | x86_64 |  
GNU/Linux |  
-----
```

```
-----  
| Kernel Version : #53~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC Wed Jan 15 19:18:46 UTC 2  
|  
-----  
-----
```

### 7.1.8. stat-info (st, stat)

Provides information about the operational state of the system and destinations in operation.

You can check dynamic information such as the current status of the destination, the names of the services being processed, the number of services processed, service status, and the number of requests waiting in the service queue.

- Usage

```
> st [-hth <HTH number>][-jeus <JEUS destination name>][-rproxy <reverse proxy destination  
name>][-htmls <HTML server name>]
```

| Option                                     | Description                                                                             |
|--------------------------------------------|-----------------------------------------------------------------------------------------|
| [-hth <HTH number>]                        | Specifies the HTH number for which destination statistics are displayed.                |
| [-jeus <JEUS destination name>]            | Specifies the JEUS destination name for which statistics are displayed.                 |
| [-rproxy <reverse proxy destination name>] | Specifies the name of the reverse proxy destination for which statistics are displayed. |
| [-htmls]                                   | Specifies whether to display statistical information.                                   |

- Example

```
[wsadmin]>> st  
Prints all information. Use options to print specific items ( -jeus, -rproxy, -htmls )  
-----
```

```
-----  
| HTH-0 : JEUS statistic  
|  
-----
```

```

-----
|           Server           | Server Group | Connections | Average Processed Time |
Request Count | Response Count | Sticky Routed Count |
-----

```

```

| amV1c19kb21haW4vYWRtaW5TZXJ2ZXI= | MyGroup1 | 10 | 0.000000 |
0 | 0 | 0 |
-----

```

```

-----
| HTH-0 : Reverse Proxy statistic
|
-----

```

```

-----
|           Server           | Server Group | Connections | Average Processed Time | Request Count |
Response Count | Sticky Routed Count |
-----

```

```

| 192.168.15.120:8088 | rproxyg1 | 0 | 0.000000 | 0 |
0 | 0 |
-----

```

```

-----
| HTH-0 : HTMLS statistic
|
-----

```

```

| Server | Queue Count | Average Processed Time | Request Count | Response Count |
| HTMLS | 0 | 0.002725 | 10 | 10 |
-----

```

The following describes each output item.

- Output items for JEUS destination statistics

| Item                   | Description                                                                      |
|------------------------|----------------------------------------------------------------------------------|
| Server                 | The name of the JEUS destination managed internally.                             |
| Server Group           | The name of the JEUS destination group.                                          |
| Connections            | The number of connections established with the JEUS destination.                 |
| Average Processed Time | The average processing time. (Unit: seconds)                                     |
| Request Count          | The number of requests sent to the destination.                                  |
| Response Count         | The number of requests processed by the destination.                             |
| Sticky Routed Count    | The number of requests sent to the destination with a Sticky ID from the client. |

- Output items for reverse proxy destination statistics

| Item                   | Description                                                                      |
|------------------------|----------------------------------------------------------------------------------|
| Server                 | The address of the reverse proxy destination.                                    |
| Server Group           | The name of the reverse proxy destination.                                       |
| Connections            | The number of connections established with the reverse proxy destination.        |
| Average Processed Time | The average processing time. (Unit: seconds)                                     |
| Request Count          | The number of requests sent to the destination.                                  |
| Response Count         | The number of requests processed by the destination.                             |
| Sticky Routed Count    | The number of requests sent to the destination with a Sticky ID from the client. |

- Output items for HTMLS destination statistics

| Item                   | Description                                  |
|------------------------|----------------------------------------------|
| Server                 | The HTMLS destination name.                  |
| Queue Count            | The number of requests waiting in HTMLS.     |
| Average Processed Time | The average processing time. (Unit: seconds) |
| Request Count          | The number of requests sent to HTMS.         |
| Response Count         | The number of requests processed by HTMLS.   |

## 7.1.9. restat

Resets destination statistics.

- Usage

```
> restat [-hth <HTH number>][-jeus <JEUS destination name>][-rproxy <reverse proxy destination name>][-htmls]
```

| Option                                     | Description                                                                              |
|--------------------------------------------|------------------------------------------------------------------------------------------|
| [-hth <HTH number>]                        | Specifies the HTH number for which destination statistics are reset.                     |
| [-jeus <JEUS destination name>]            | Specifies the JEUS destination name for which statistical information is reset.          |
| [-rproxy <reverse proxy destination name>] | Specifies the reverse proxy destination name for which statistical information is reset. |
| [-htmls]                                   | Specifies whether to reset HTMLS destination statistics.                                 |

- Example

Using this command without any options resets all statistical information.

```
[wsadmin]>> restat
Prints all information. Use options to print specific items ( -jeus, -rproxy, -htmls )
-----
| HTH-0 |
-----
| List of JEUS with cleared statistical information |
-----
| MyGroup2 |
| MyGroup1 |
-----

-----
| HTH-0 |
-----
| List of Reverse Proxy with cleared statistical information|
-----
| rproxyg1 |
-----

-----
| HTH-0 |
-----
| List of HTMLS with cleared statistical information|
-----
| [HTMLS] |
-----
```

### 7.1.10. config (cfg)

Displays the environment information of the currently running system. It allows you to check all environment information including default values by nodes, server groups, servers, destinations, and services defined in the environment configuration file.

- Usage

```
> config [-path <JSON path>]
```

| Option              | Description                                                                                                                                                                                            |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| [-path <JSON path>] | Specifies the JSON path to the configuration file to check. The path must always start with '/'.<br>If the path is omitted or set to '/', all currently set environment information will be retrieved. |

- Example

The following is an example of displaying the environment configuration for the NODE section. For detailed information about the configuration items in the NODE section, refer to

```
[wsadmin]>> cfg -path /node
-----
| Target Config /node                               |
-----
|                               Values              |
-----
| {                                                 |
|   "cache_entry": 128,                             |
|   "cache_key": "HOST_URI",                         |
|   "cache_max_file_size": 8192,                     |
|   "connection_pool_size": 8192,                   |
|   "graceful_shutdown_timeout": 60,                 |
|   "hth_count": 1,                                 |
|   "hth_schedule": "RR",                           |
|   "limit_request_body_size": 0,                    |
|   "limit_request_header_field_count": 100,         |
|   "limit_request_header_field_size": 8190,         |
|   "limit_request_line_size": 8190,                 |
|   "listen_backlog": 4096,                          |
|   "max_cache_memory_size": 100,                    |
|   "name": "ksh",                                   |
|   "system_filters": [],                            |
|   "worker_threads": 1                             |
| }                                                 |
-----
```

### 7.1.11. update-config

Reloads the configuration file and applies changes dynamically. Only settings specified with the **dynamic** key in the schema file are applied.

- Usage

This command does not require any separate options.

### 7.1.12. cache-list (cachelist)

Displays information about the responses stored in WebtoB's HTTP response cache.

- Usage

```
> cachelist [-hth <HTH number>]
```

| Option              | Description                                                                                                               |
|---------------------|---------------------------------------------------------------------------------------------------------------------------|
| [-hth <HTH number>] | Specifies the HTH for which to retrieve cache information. If not specified, cache information for all HTHs is retrieved. |

- Example

The following is an example of a cached response to the request "/test.txt". The rest of the request path is used only for internal server debugging.

```
[wsadmin]>> cachelist
-----
| HTH-0: Cache List Info |
-----
|      Cache key      |   Expired time   | Cache size |
-----
| 192.168.15.120:8080/test.txt | 2024-10-29 12:34:53 |      286 |
-----
| Cache count : 1 |
| Memory usages : 286 |
-----
```

### 7.1.13. cache-refresh (cr)

Removes the response data from the WebtoB HTTP response cache.

- Usage

```
> cacherefresh { [-url <URL>] | [-ext <extension>] }
```

| Option             | Description                                         |
|--------------------|-----------------------------------------------------|
| [-url <URL>]       | Specifies the URL of the cache to be removed.       |
| [-ext <extension>] | Specifies the extension of the cache to be removed. |

- Example

Using this command without any options removes all stored cache information.

```
[wsadmin]>> cr
[HTH-0] Number of removed cache items: 10
```

### 7.1.14. cert-refresh (cr)

Refreshes WebtoB's certificate information.

- Usage

This command does not require any separate options.

## 7.1.15. svg-info (si, sgi, svginfo)

Displays information about the destination that processes the client's request.

- Usage

```
> si [-htmls][-jeus <JEUS destination name>][-rproxy <reverse proxy destination name>]
```

| Option                                     | Description                                                                      |
|--------------------------------------------|----------------------------------------------------------------------------------|
| [-htmls]                                   | Specifies whether to display the HTMLS destination information.                  |
| [-jeus <JEUS destination name>]            | Specifies the JEUS destination name for which information is displayed.          |
| [-rproxy <reverse proxy destination name>] | Specifies the reverse proxy destination name for which information is displayed. |

- Example

```
[wsadmin]>> si
-----
| HTH-0 : HTMLS Destination Info                                     |
-----
| svgname | status | aqcnt | count | cqcnt | qpcent | reqs | rsent |
-----
| HTMLS   | RDY    | 0     | 0     | 0     | 0      | 0    | 0     |
-----

-----
| HTH-0 : JEUS Destination Info                                     |
-----
| svgname | status | aqcnt | count | cqcnt | qpcent | reqs | rsent |
-----
| MyGroup2 | NRDY   | 0     | 0     | 0     | 0      | 0    | 0     |
| MyGroup1 | RDY    | 0     | 0     | 0     | 0      | 0    | 0     |
-----

-----
| HTH-0 : Reverse Proxy Destination Info                           |
-----
| svgname | status | aqcnt | count | cqcnt | qpcent | reqs | rsent |
-----
| rproxyg1 | RDY    | 0     | 0     | 0     | 0      | 0    | 0     |
-----
```

The following describes each output item.

| Item    | Description       |
|---------|-------------------|
| svgname | Destination name. |

| Item   | Description                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| status | Client status within the destination. <ul style="list-style-type: none"> <li>◦ RDY: The destination can process requests and there are destinations connected to WebtoB.</li> <li>◦ NRDY: The destination cannot process requests and there are no destinations connected to WebtoB.</li> <li>◦ BLK: The destination has been temporarily suspended by an administrator command and cannot process requests.</li> </ul> |
| aqcnt  | Cumulative number of queued requests. (Cumulative cqcnt value)                                                                                                                                                                                                                                                                                                                                                          |
| count  | Number of processed requests.                                                                                                                                                                                                                                                                                                                                                                                           |
| cqcnt  | Number of requests in the queue.                                                                                                                                                                                                                                                                                                                                                                                        |
| qpcnt  | Number of queued requests that have been removed from the queue due to request timeout, qp command, etc.                                                                                                                                                                                                                                                                                                                |
| reqs   | Number of requests sent to the destination.                                                                                                                                                                                                                                                                                                                                                                             |
| rscnt  | Number of destination restarts due to abnormal termination.                                                                                                                                                                                                                                                                                                                                                             |

## 7.1.16. connection-info (ci, conl, coninfo)

Displays connection information between WebtoB's clients and destinations.

- Usage

```
> ci [-jeus <JEUS destination name>][-rproxy <reverse proxy destination name>]
```

| Option                                     | Description                                                                      |
|--------------------------------------------|----------------------------------------------------------------------------------|
| [-jeus <JEUS destination name>]            | Specifies the JEUS destination name for which information is displayed.          |
| [-rproxy <reverse proxy destination name>] | Specifies the reverse proxy destination name for which information is displayed. |

- Example

```
[wsadmin]>> ci
-----
| HTH-0 : Connection info
|
-----
| No | Server | Local Address | Remote Address | Remote Type | Ssl | Status | Request
Count | Response Count | Idle Time | Mapping No |
-----
```

|    |       |                     |                      |        |    |       |   |
|----|-------|---------------------|----------------------|--------|----|-------|---|
| 0  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40158 | JSV    | No | READY | 1 |
| 1  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40170 | JSV    | No | READY | 1 |
| 2  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40186 | JSV    | No | READY | 1 |
| 3  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40202 | JSV    | No | READY | 1 |
| 4  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40214 | JSV    | No | READY | 1 |
| 5  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40220 | JSV    | No | READY | 1 |
| 6  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40224 | JSV    | No | READY | 0 |
| 7  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40230 | JSV    | No | READY | 0 |
| 8  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40234 | JSV    | No | READY | 0 |
| 9  | WJP-0 | 192.168.15.120:9999 | 192.168.15.120:40240 | JSV    | No | READY | 0 |
| 21 | http1 | 192.168.15.120:8080 | 192.168.15.120:47098 | CLIENT | No | READY | 3 |

The following describes each output item.

| Item                             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| No                               | The connection number managed internally by WebtoB.                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Server                           | The server name managed internally by WebtoB.                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Local Address,<br>Remote Address | IP address and port number of the server and client.                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Remote Type                      | <p>The type of connection.</p> <ul style="list-style-type: none"> <li>◦ CLIENT: Connection with the client</li> <li>◦ CLIENT_WEBSOCKET: Websocket connection with the client</li> <li>◦ RPROXY: Connection with an internal reverse proxy destination</li> <li>◦ RPROXY_PERSISTENT: Keep-Alive connection with an internal reverse proxy destination</li> <li>◦ RPROXY_WEBSOCKET: Websocket connection with an internal reverse proxy destination</li> <li>◦ JSV: Connection with an internal JEUS destination</li> </ul> |
| Ssl                              | Specifies whether the connection is established using SSL.                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Status                           | <p>The status of the connection.</p> <ul style="list-style-type: none"> <li>◦ READY: Ready to receive requests from clients.</li> <li>◦ RUNNING: Processing the client's request.</li> </ul>                                                                                                                                                                                                                                                                                                                              |

| Item           | Description                                                                                                                                                                                                                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Request Count  | The number of requests sent through this connection.                                                                                                                                                                                                                                                                 |
| Response Count | The number of requests processed through this connection.                                                                                                                                                                                                                                                            |
| Idle Time      | Idle time with no data exchanges.                                                                                                                                                                                                                                                                                    |
| Mapping No     | The connection number associated with this connection.<br>If it is a client connection, it displays the destination connection number handling the request. If it is a destination connection, it displays the client connection number that sent the request. If there is no associated connection, it displays -1. |

### 7.1.17. shutdown (down)

Shuts down the running WebtoB.

- Usage

```
> shutdown [-t <forced shutdown time>]
```

| Option                      | Description                                                                                               |
|-----------------------------|-----------------------------------------------------------------------------------------------------------|
| [-t <forced shutdown time>] | If WebtoB does not shut down within the specified time after the command, it will be forcibly terminated. |

### 7.1.18. exit

Exits the wsadmin tool.

- Usage

```
[wsadmin]>> exit
```

## 7.2. configMigrator

Migrates WebtoB 5 configuration files to WebtoB 6.

Internally, the configMigrator compiles WebtoB 5 configuration file into an intermediate file (wsconfig) and then performs migration based on the compilation results.



By default, all files related to configMigrator are based on the config directory unless a separate path is specified.

Therefore, the wsconfig file is also created in the config directory by default.

- Usage

```
$ configMigrator [-h|--help] [-v|--version] [-i|--input_config_name <WebtoB5 config file name>]
[-o|--output_config_name <WebtoB6 config file name>] [-f|--format <WebtoB6 config file
extension>]
```

| Option                                                | Description                                                                                                                                                                                                                                            |
|-------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| [-h --help]                                           | Displays descriptions of options available in configMigrator.<br><br>※ Any other options used with the <b>-h</b> option are ignored.                                                                                                                   |
| [-v --version]                                        | Displays the WebtoB version.<br><br>※ Any other options used with the <b>-v</b> option are ignored except for the <b>-h</b> option).                                                                                                                   |
| [-i --input_config_name <WebtoB 5 config file name>]  | Specifies the WebtoB 5 configuration file.<br><br>Relative or absolute paths can be specified based on the config directory. (Default: http.m)                                                                                                         |
| [-o --output_config_name <WebtoB 6 config file name>] | Specifies the WebtoB 6 configuration file.<br><br>Relative or absolute paths can be specified based on the config directory. (Default: webtob-config-generated)                                                                                        |
| [-f --format <WebtoB 6 config file extension>]        | Specifies the extension for the WebtoB 6 configuration file. (Default: json) <ul style="list-style-type: none"><li>◦ json</li><li>◦ yml</li><li>◦ yaml</li></ul> ※ This option can be omitted if the extension is specified with the <b>-o</b> option. |

- Example

No Options Used (when the http.m file exists in the config directory)

```
$ configMigrator
    Environment variables for WebtoB6:
        WEBTOB_HOME = /home/tmax/webtob6/

Current configuration:
    Number of client handler(HTH) = 1
    Supported maximum user per node = 16357
```

```
Supported maximum user per handler = 16357
```

```
Successfully created the configuration file (/home/tmax/webtob6/config/wsconfig) for node webtob1.
```

```
The host name of the running machine is tmax.
```

```
legacy config file: /home/tmax/webtob6/config/http.m
```

```
modern config file: /home/tmax/webtob6/config/webtob-config-generated.json
```

```
migration complete: /home/tmax/webtob6/config/webtob-config-generated.json
```

### Specifying input\_config\_name and output\_config\_name with Paths

```
$ configMigrator -i ~/webtob5/config/http.m -o ~/webtob6/config/webtob-config-migrated.yaml
```

```
Environment variables for WebtoB6:
```

```
WEBTOB_HOME = /home/tmax/webtob6/
```

```
Current configuration:
```

```
Number of client handler(HTH) = 1
```

```
Supported maximum user per node = 16337
```

```
Supported maximum user per handler = 16337
```

```
Successfully created the configuration file (/home/tmax/webtob6/config/wsconfig) for node webtob1.
```

```
The host name of the running machine is tmax.
```

```
legacy config file: /home/tmax/webtob5/config/http.m
```

```
modern config file: /home/tmax/webtob6/config/webtob-config-migrated.yaml
```

```
migration complete: /home/tmax/webtob6/config/webtob-config-migrated.yaml
```

## 7.3. configValidator

The configValidator performs schema validation on WebtoB configuration files.

Before running WebtoB, you can check the verification results for the configuration file using the configValidator tool.



WebtoB's default configuration file is webtob-config.json, and you can change it by setting the environment variable WEBTOB\_CONFIG\_FILE\_NAME.

- Usage

```
$ configValidator
```

- Example

## Validating a Normal WebtoB Configuration File

```
$ configValidator
Config file path: ../config/webtob-config.json
Schema file path: ../config/webtob-config.schema.json
WEBTOB_HOME = ../
***** Start config validation *****
1. Opened schema file
2. Opened config file
3. Config parsing finished
4. Pre-validation process finished
5. Validation using json-schema finished
6. Post-validation process finished
Config file "../config/webtob-config.json" is validated with schema file "../schema/webtob-
config.schema.json"
Success to load config files : webtob-config.json
***** Validation success *****
```

## Validating an Abnormal WebtoB Configuration File

```
$ configValidator
Config file path: ../config/webtob-config.json
Schema file path: ../config/webtob-config.schema.json
WEBTOB_HOME = ../
***** Start config validation *****
1. Opened schema file
2. Opened config file
3. Config parsing finished
4. Pre-validation process finished
[Exception on json-schema validation][Config validation failed]At /node/hth_count of -1 -
instance is below minimum of 1
```

## 7.4. mkpwd

mkpwd is a tool to support certificate\_key\_password in the SSL section.

When an encrypted private key is set in the SSL section, WebtoB prompts you to enter a passphrase every time it starts. To avoid repeatedly entering the passphrase, you can set the certificate\_key\_password. The mkpwd tool creates a passphrase file that stores the passphrase so that it can be applied to the certificate\_key\_password.



For more information on using certificate\_key\_password, refer to [Configuration Items](#).

- Usage

```
$ mkpwd <file path> <SSL name>
```

| Options     | Description                                                                                            |
|-------------|--------------------------------------------------------------------------------------------------------|
| <file path> | Name of the file used for certificate_key_password. The result of running mkpwd is saved to this file. |
| <SSL name>  | The name set in the SSL section.                                                                       |

- Example

No Options Used

```
$ mkpwd
<< Usage >>
$ mkpwd file_path ssl_name
    file_path: output file for ssl certificate key password
    ssl_name: name of SSL section
```

Specifying file\_path and ssl\_name

```
$ mkpwd ssl.ppd ssl1
Make password for SSL certificate key password
Enter password: (Enter password)
Successfully Added password for [ssl1] to a file [ssl.ppd].

$ls -al ssl.ppd
-rw-rw-r-- 1 webtob webtob 14 Nov 6 12:34 ssl.ppd
```

- Verifying File Contents

```
$ cat ssl.ppd
ssl1 dGVzdA==
```

# Appendix A: Configuration File Examples

This appendix provides examples of basic configuration settings, JEUS integration configuration settings, and commonly used configuration settings.

## A.1. Basic Configuration

The following is an example of a basic configuration file for WebtoB.

```
{
  "node": {
    "name": "mynode"
  },
  "server": {
    "http": {
      "common_config": {
        "doc_root": "docs/",
        "access_log": "accesslog",
        "alias": [
          "alias"
        ]
      },
      "http_servers": [
        {
          "name": "http1",
          "port": [
            8080
          ]
        }
      ]
    }
  },
  "service": {
    "uri": [
      {
        "name": "uri",
        "match": {
          "target": "/htmls/",
          "rewrite": "/"
        },
        "destination": {
          "type": "HTMLS",
          "target": "#default_htmls"
        }
      }
    ],
    "ext": [
      {
        "name": "htm",
        "match": {
          "target": "htm"
        },
        "destination": {
          "type": "HTMLS",
```

```

        "target": "#default_html5"
    }
}
],
"logging": {
    "system_log": [
        {
            "name": "webtob",
            "handlers": {
                "file_handler": {
                    "file_name": "log/system_%Y%M%D%.log"
                }
            }
        }
    ],
    "access_log": [
        {
            "name": "accesslog",
            "format": "%h %t \"%r\" %s %b %D",
            "handlers": {
                "file_handler": {
                    "file_name": "log/access_%Y%M%D%.log"
                }
            }
        }
    ],
    "error_log": {
        "format": "%r",
        "handlers": {
            "file_handler": {
                "file_name": "log/error_%Y%M%D%.log"
            }
        }
    }
}
}
}

```

## A.2. JEUS Integration Configuration

### A.2.1. WebtoB Configuration

The following is an example of a WebtoB configuration file for integration with JEUS.

```

{
    "node": {
        "name": "mynode"
    },
    "server": {
        "http": {
            "common_config": {
                "doc_root": "docs/",
                "access_log": "accesslog"
            },

```

```

    "http_servers": [
      {
        "name": "http1",
        "port": [
          8080
        ]
      }
    ],
    "wjp": {
      "wjp_servers": [
        {
          "name": "MyGroup",
          "port": 9999
        }
      ]
    },
    "destination": {
      "jeus": [
        {
          "name": "MyGroup"
        }
      ]
    },
    "service": {
      "uri": [
        {
          "name": "examples",
          "match": {
            "target": "/examples/"
          },
          "destination": {
            "type": "JEUS",
            "target": "MyGroup"
          }
        }
      ],
      "ext": [
        {
          "name": "jsp",
          "match": {
            "target": "jsp"
          },
          "destination": {
            "type": "JEUS",
            "target": "MyGroup"
          }
        },
        {
          "name": "htm",
          "match": {
            "target": "htm"
          },
          "destination": {
            "type": "HTMLS",
            "target": "#default_htmls"
          }
        }
      ],
    },

```

```

    {
        "name": "html",
        "match": {
            "target": "html"
        },
        "destination": {
            "type": "HTMLS",
            "target": "#default_htmls"
        }
    }
]
},
"logging": {
    "system_log": [
        {
            "name": "webtob",
            "handlers": {
                "file_handler": {
                    "file_name": "log/system_%Y%M%D.log"
                }
            }
        }
    ],
    "access_log": [
        {
            "name": "accesslog",
            "format": "%h %t \"%r\" %s %b %D",
            "handlers": {
                "file_handler": {
                    "file_name": "log/access_%Y%M%D.log",
                    "enable_sync": true
                }
            }
        }
    ],
    "error_log": {
        "format": "%r",
        "handlers": {
            "file_handler": {
                "file_name": "log/error_%Y%M%D.log"
            }
        }
    }
}
}
}

```

## A.2.2. JEUS Configuration

The following is an example of a JEUS configuration file for integration with WebtoB.

The following example is based on JEUS 9. Integration with WebtoB is configured in `<webtob-connector>`.

domain.xml

```
<?xml version="1.0" encoding="UTF-8"?><domain xmlns="http://www.tmaxsoft.com/xml/ns/jeus"
```

```

version="9.0">
<webtob-connector>
  <name>webtob-connector</name>
  <thread-pool>
    <min>5</min>
    <max>5</max>
    <max-idle-time>300000</max-idle-time>
    <max-queue>-1</max-queue>
    <thread-state-notify>
      <max-thread-active-time>0</max-thread-active-time>
      <interrupt-thread>false</interrupt-thread>
      <active-timeout-notification>false</active-timeout-notification>
      <notify-threshold-ratio>0.0</notify-threshold-ratio>
      <restart-threshold-ratio>0.0</restart-threshold-ratio>
    </thread-state-notify>
  </thread-pool>
  <postdata-read-timeout>30000</postdata-read-timeout>
  <max-post-size>-1</max-post-size>
  <max-parameter-count>-1</max-parameter-count>
  <max-header-count>-1</max-header-count>
  <max-header-size>8192</max-header-size>
  <max-querystring-size>8192</max-querystring-size>
  <wjp-version>2</wjp-version>
  <registration-id>MyGroup</registration-id>
  <hth-count>1</hth-count>
  <connection-count>5</connection-count>
  <threshold>-1</threshold>
  <use-nio>false</use-nio>
  <network-address>
    <port>9999</port>
    <ip-address>localhost</ip-address>
  </network-address>
  <read-timeout>120000</read-timeout>
  <reconnect-interval>5000</reconnect-interval>
  <reconnect-count-for-backup>12</reconnect-count-for-backup>
  <request-prefetch>false</request-prefetch>
</webtob-connector>

```

## A.3. Commonly Used Settings

The following is an example of a configuration file containing frequently used configuration settings for WebtoB.

```

{
  "node": {
    "name": "mynode",
  },
  "server": {
    "http": {
      "common_config": {
        "doc_root": "docs/",
        "error_document": [
          "400",
          "401",
          "403",

```

```

        "404",
        "405",
        "503"
    ],
    "access_log": "acc_node",
    "enable_allow_header_on_405": false
},
"http_servers": [
    {
        "name": "http1",
        "vhosts": [
            {
                "name": "v_tmaxsoft",
                "common_config": {
                    "doc_root": "docs/",
                    "error_document": [
                        "400",
                        "401",
                        "403",
                        "404",
                        "405",
                        "503"
                    ],
                },
                "access_log": "acc_tmaxsoft",
                "enable_url_rewrite": true,
                "url_rewrite_config": "config/rewrite.conf",
                "index_name": "blank.html"
            },
            {
                "host_name": [
                    "tmaxsoft.com",
                    "www.tmaxsoft.com"
                ]
            }
        ]
    },
    {
        "name": "http2",
        "enable_ssl": true,
        "ssl_name": [
            "ssl_tmaxsoft"
        ],
        "bind_ipv6_only": true,
        "vhosts": [
            {
                "name": "v_tmaxsoft_ssl",
                "common_config": {
                    "doc_root": "docs/",
                    "error_document": [
                        "400",
                        "401",
                        "403",
                        "404",
                        "405",
                        "503"
                    ],
                },
                "access_log": "acc_tmaxsoft_ssl",
                "index_name": "blank.html"
            },
            {
                "host_name": [

```

```

        "tmaxsoft.com",
        "www.tmaxsoft.com"
    ]
    }
}
],
"mime_type_file": "config/mime.types"
},
"wjp": {
    "port": 9900,
    "wjp_servers": [
        {
            "name": "tmaxsoft"
        }
    ]
},
},
"destination": {
    "jeus": [
        {
            "name": "tmaxsoft"
        }
    ],
    "reverse_proxy": {
        "reverse_proxy_group": [
            {
                "name": "rp_rp_tmaxsoft",
                "reverse_proxy_server": [
                    {
                        "name": "rp_rp_tmaxsoft",
                        "address": "localhost:8443",
                        "enable_proxy_ssl": true,
                        "proxy_ssl_name": [
                            "pssl_v0"
                        ],
                    },
                    {
                        "common_config": {
                            "server_health_check": {
                                "enable_name_resolution_on_fail": false
                            }
                        }
                    }
                ]
            }
        ]
    },
    "htmls": [
        {
            "name": "#default_htmls"
        }
    ]
},
"ssl": {
    "ssl_configs": [
        {
            "name": "ssl_tmaxsoft",
            "certificate_file": "ssl/cert.crt",
            "certificate_key_file": "ssl/privkey.key",
            "certificate_chain_file": "ssl/chain.pem",

```

```

        "ca_certificate_file": "ssl/rootca.pem",
        "ca_certificate_path": "ssl/",
        "common_config": {
            "protocols": [
                "TLSv1.2",
                "TLSv1.3"
            ],
            "required_ciphers": "HIGH:!RSA:!SHA1"
        }
    },
    ],
    "proxy_ssl_configs": [
        {
            "name": "pssl_v0",
            "enable_insecure": true
        }
    ]
},
"service": {
    "uri": [
        {
            "name": "u_tmaxsoft_root",
            "target_http_servers": [
                "http1.v_tmaxsoft",
                "http2.v_tmaxsoft_ssl"
            ],
            "match": {
                "target": "/",
                "goto_ext": true
            },
            "destination": {
                "type": "JEUS",
                "target": "tmaxsoft"
            }
        },
        {
            "name": "rp_rp_tmaxsoft",
            "target_http_servers": [
                "http1.v_tmaxsoft",
                "http2.v_tmaxsoft_ssl"
            ],
            "match": {
                "rewrite": "/ws/",
                "target": "/ws/"
            },
            "destination": {
                "type": "REVERSE_PROXY",
                "target": "rp_rp_tmaxsoft"
            }
        },
        {
            "name": "u_tmaxsoft_error",
            "target_http_servers": [
                "http1.v_tmaxsoft",
                "http2.v_tmaxsoft_ssl"
            ],
            "match": {
                "target": "/error"
            }
        }
    ]
},

```

```

        "destination": {
            "type": "HTMLS",
            "target": "#default_htmls"
        }
    },
    ],
    "ext": [
        {
            "name": "jsp",
            "match": {
                "target": "jsp"
            },
            "destination": {
                "type": "JEUS",
                "target": "tmaxsoft"
            }
        }
    ]
},
"logging": {
    "system_log": [
        {
            "name": "webtob",
            "handlers": {
                "file_handler": {
                    "file_name": "log/system/system_%Y%M%D.log",
                    "enable_sync": true
                }
            }
        }
    ],
    "access_log": [
        {
            "name": "acc_node",
            "format": "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\"",
            "handlers": {
                "file_handler": {
                    "file_name": "log/node/access_%Y%M%D.log",
                    "enable_sync": true
                }
            }
        },
        {
            "name": "acc_tmaxsoft",
            "format": "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\"",
            "handlers": {
                "file_handler": {
                    "file_name": "log/tmaxsoft/access_%Y%M%D.log",
                    "enable_sync": true
                }
            }
        },
        {
            "name": "acc_tmaxsoft_ssl",
            "format": "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\"",
            "handlers": {
                "file_handler": {
                    "file_name": "log/tmaxsoft_ssl/access_%Y%M%D.log",
                    "enable_sync": true
                }
            }
        }
    ]
}

```

```

    }
  }
},
"error_log": {
  "format": "%r",
  "handlers": {
    "file_handler": {
      "file_name": "log/node/error_%Y%M%D%.log",
      "enable_sync": true
    }
  }
},
},
"error_document": {
  "error_document_list": [
    {
      "name": "400",
      "status": 400,
      "url": "/error/error.html"
    },
    {
      "name": "401",
      "status": 401,
      "url": "/error/error.html"
    },
    {
      "name": "403",
      "status": 403,
      "url": "/error/error.html"
    },
    {
      "name": "404",
      "status": 404,
      "url": "/error/error.html"
    },
    {
      "name": "405",
      "status": 405,
      "url": "/error/error.html"
    },
    {
      "name": "503",
      "status": 503,
      "url": "/error/error.html"
    }
  ]
},
"headers": {
  "headers_list": [
    {
      "name": "hsts",
      "action": "AddResponse",
      "field_name": "Strict-Transport-Security",
      "field_value": "max-age=2592000; includeSubDomains"
    },
    {
      "name": "xcontent",
      "action": "AddResponse",

```

```

        "field_name": "X-Content-Type-Options",
        "field_value": "nosniff"
    },
    {
        "name": "xframe",
        "action": "AddResponse",
        "field_name": "X-Frame-Options",
        "field_value": "SAMEORIGIN"
    },
    {
        "name": "xxss",
        "action": "AddResponse",
        "field_name": "X-XSS-Protection",
        "field_value": "1; mode=block"
    },
    {
        "name": "cachectrl",
        "action": "AddResponse",
        "field_name": "Cache-Control",
        "field_value": "no-cache"
    }
]
}

```