



OPC-HTTP Gateway

<b>1</b>		<b>1</b>
<b>2</b>	,	<b>1</b>
1	.....	1
2	.....	2
3	.....	2
4	.....	3
<b>3</b>		<b>3</b>
1	.....	3
2	.....	3
<b>4</b>		<b>3</b>
1	.....	3
2	.....	4
3	.....	5
4 HTTP-	.....	5
5	SSL .....	7
6 WebSocket	.....	7
7	OPC UA .....	8
<b>5</b>		<b>8</b>
<b>6 HTTP-</b>		<b>9</b>
1	.....	9
2	.....	10
3	.....	10
4	JSON .....	11
5	.....	12
6	.....	12
7	.....	14
8	.....	15
9	.....	16
10	.....	16
11	.....	18
12	.....	18



# 1

HTTP-  
HTTP(S) , OPC- OPC-  
(DA1, DA2, DA3 UA).  
OPC-

## 2

### 2.1

Copyright © 1999-2025 AGG Software.  
All Rights Reserved

1. OPC-HTTP Gateway
2. OPC-HTTP Gateway  
sales@aggsoft.ru
3. OPC-HTTP Gateway  
OPC-HTTP Gateway
4. **OPC-HTTP Gateway** "AS IS".
- 5.
6. OPC-HTTP Gateway

7. HTTP Gateway  
HTTP Gateway.

OPC-  
OPC-

## 2.2

shareware-

- 21
- 
- 65535

## 2.3

## 2.4

OPC-HTTP Gateway	<a href="https://www.aggsoft.ru/support/">https://www.aggsoft.ru/support/</a>
------------------	---

## 3

### 3.1

- Windows 2000 SP 4 - Windows 11 ( x86 x64, ).

, ( , ), , COM- , .

### 3.2

OPC-HTTP Gateway.

Gateway" , OPC-HTTP Gateway "Programs Files\OPC-HTTP  
 OPC-HTTP Gateway

## 4

### 4.1

opchttpgate.xml  
 opchttpgate.exe. OPC UA  
 opcuahtpgate.xml, opcuahtpgate.exe

HTTP-

```

:

<Config>
  <Log file="opchttpgate.log" info="1" warning="1" error="1" limit="1000000" />
  <HTTPServer port="80;443" ssl="0" auth="1" staticfilepath="files\" log="1"
logdetailed="0">
  <Users>
    <User login="admin" password="admin" />
    <User login="user" password="user" />
  </Users>
  <AllowedIPs>
    <IP>127.0.0.*</IP>
    <IP>192.168.1.*</IP>
  </AllowedIPs>
</HTTPServer>
</Config>

```

## 4.2

"Log" XML

```

<Log file="opchttpgate.log" info="1" warning="1" error="1" limit="1000000" />

```

**file** ( ; : ) -

**info** ( ; : 1) -  
 "1" ( ),

**warning** ( ; : 1) -  
 info,

**error** ( ; : 1) -  
 info,

**limit** ( ; : 100000) -

### 4.3

"DesktopModeSettings" XML

:

```
<DesktopModeSettings>
  <CustomTitle addversion="1">File OPC Server</CustomTitle>
  <CustomLabel>Uptime</CustomLabel>
  <ExitPassword enabled="1">admin</ExitPassword>
  <StopPassword enabled="1">admin</StopPassword>
</DesktopModeSettings>
```

**CustomTitle** ( ; : ) -  
"addversion" "1",

**CustomLabel** ( ; : ) -

**ExitPassword** ( "enabled" ; "1", : ) -  
"blockuserlogout" "1",

**StopPassword** ( "enabled" ; "1", : ) -

### 4.4 HTTP-

"File" XML

HTTP-

:

```
<HTTPServer port="80;443" ssl="0" auth="1" staticfilepath="files\" log="1"
logdetailed="0">
```

**port** ( ; : 80) - TCP- TCP- .  
(«;»). HTTP-

**ssl** ( «1» ( ), ; : 0) -  
«0», 443, HTTP- SSL.  
SSL.

**websocket** ( `<1>` ( ), ; : 0) - [WebSocket](#) ;

**auth** ( `<1>` ( ), ; : 0) - HTTP- (Basic Authentication). Users HTTP- «

**staticfilepath** ( ; : files\ ) - GET.

**log** ( `<1>` ( ), HTTP- ; : 1) -

**logdetailed** - `<1>` ( ), HTTP-

### (Users)

```
:
<User login="admin" password="admin" />
```

### IP- (AllowedIPs)

```
IP- , HTTP-
IP-
IP- «?» «*».
:
192.168.1.* - IP- 192.168.1.0 192.168.1.255
```

### AllowedOPC

```
OPC
,
"?" "*"
server_id ( ) -
host ( : ) - IP- OPC- OPC HTTP.
```

```

:
<AllowedOPC>
  <!-- remove all "OPC" nodes if you want to allow access for all. Mask characters *
  and ? are allowed -->
  <OPC server_id="opcserver Instance.*" host="*">
    <User login="*" />
  </OPC>
</AllowedOPC>

```

«opcserver Instance».

OPC-

## 4.5

## SSL

HTTP-  
SSL

SSL,

01ServerCert.pem -  
01ServerKey.pem -

127.0.0.1.

OpenSSL.

SSL

## 4.6 WebSocket

"demo\_websocket.html",

JSON.

[HTTP](#)

## 4.7

## OPC UA

OPC-HTTP Gateway

OPC UA

- None ( )
- Sign ( )
- Sign & Encrypt ( )

- Basic128Rsa15
- Basic256
- Basic256Sha256

OPC-HTTP Gateway

OPC UA  
"None".

OPC-HTTP Gateway

( ) OPC

Application Uri:urn:aggsoft.com.opc.client.application

Serial Number: 00f8e216f16b422e72

Signature algorithm: sha256RSA

Issuer: CN = aggsoft.com.opc.client@localhost;O = aggsoft.com;C = DE

Sucject: CN = aggsoft.com.opc.client@localhost;O = aggsoft.com;C = DE

Thumbprint: 2603744fb26c3345f6880e451fe76b9e19c0

## 5

Windows.

(cmd.exe)

cmd.exe

Windows Vista

: opchttpgate.exe /AI

: opchttpgate.exe /R

"OPC HTTP Gateway Service"

(cmd.exe)

ophttpgate.exe /?

## 6 HTTP-

### 6.1

#### HTTP

1. HTTP- GET POST.
2. POST: «application/x-www-form-urlencoded».
3. POST POST, HTTP-  
«Post Data»,
4. HTTP- JSON  
Content-Type «application/json; charset=utf-8».
5. ( , -  
) , HTTP- 400 (Bad Request - )).
6. ( ,  
) , HTTP- 403 (Access Denied -  
) ).
7. HTTP- .

#### WebSocket

1. HTTP- WebSocket.
2. JSON, UTF-8.
3. HTTP- (
4. HTTP- , JSON.
5. HTTP-  
(subscribe) OPC , ,

## 6.2

(write)

13	
1	Null ( )
2	Smallint ( )
3	Integer ( )
4	Single ( )
5	Double ( )
6	
7	
8	
11	
16	ShortInt ( )
17	
18	
19	
20	Int64
21	Word64

## 6.3

OPC

OPC

( ):

0x00 -

(BAD).

0x40 -

(UNCERTAIN).

0xC0 -

(GOOD).



```

clsid - OPC- .
desc - OPC- .
path - ( OPC UA) OPC .
          "\u000D" ASCII CR.

```

## 6.5

```

          OPC- .
: get-servers.json
:
host ( , ) - IP- .
      HTTP-
      HTTP:
http://127.0.0.1/get-servers.json?host=192.168.1.180
http://127.0.0.1/get-servers.json
      WebSocket JSON:
{"command": "get-servers", "host": "192.168.1.180"}
{"command": "get-servers"}
:
{"command": "get-servers", "success": true, "processed": 1, "data":
[{"name": "opcserver sim.Instance.1", "desc": "Demo server", "clsid": "{0a9db5fd-
4ca5-4611-84fb-3f5b75692fa9}", }]}
OPC UA: discovery-
          Local Discovery Server OPC Foundation

```

## 6.6

```

          OPC- .
: get-tag-tree.json
:

```

**server** ( , ) - OPC-

**host** ( , ) - IP- OPC-  
OPC UA ,

HTTP:

http://127.0.0.1/get-tag-tree.json?server=opcserver.sim.Instance.1

WebSocket JSON:

```
{"command": "get-tag-tree", "server": "opcserver.sim.Instance.1"}
```

WebSocket JSON OPC UA:

```
{"command": "get-tag-tree",  
"server": "opc.tcp://192.168.1.3:16664/OpenOpcUaCoreServer"}
```

OPC DA:

```
{"command": "get-tag-tree", "success": true, "data":  
[{"name": "Channel1.Device1.Tag1", "data_type": 8},  
{"name": "Channel1.Device1.Tag2", "data_type": 3},  
{"name": "Channel1.Device1.Tag3", "data_type": 3},  
{"name": "Channel2.Device1.Tag1", "data_type": 3}]}
```

OPC UA:

```
{"command": "", "success": true, "data":  
[{"name": "NS4|string|  
ByteString", "display_name": "ByteString", "data_type": 0, "path": "NonUaNodeComplianceTest"},  
{"name": "NS4|string|  
SByte", "display_name": "SByte", "data_type": 0, "path": "NonUaNodeComplianceTest"},  
{"name": "NS4|string|  
Time", "display_name": "Time", "data_type": 0, "path": "NonUaNodeComplianceTest"},  
{"name": "NS4|string|  
String", "display_name": "String", "data_type": 0, "path": "NonUaNodeComplianceTest"}]}
```

## OPC UA

OPC

(Node Id)

OPC

1. ( , NS4).
2. (number, string, octet, guid).
3. ( , ByteString).

**"path"**

"\".

## 6.7

```

poll_rate ( ) OPC
: subscribe.json
:
server ( , ) - OPC
host ( , ) - IP- OPC
group ( OPC- OPC- ) -
tag ( , ) - ( ) OPC-
Get Tag Tree.
poll_rate ( , ) -
OPC- , 1000 , OPC- 200

```

OPC DA

« » («|»).

OPC UA

(\u0009) (\u000D).

HTTP:

<http://127.0.0.1/subscribe.json?server=opcserverim.Instance.1&group=MyGroup1&tag=Channel1.Device1.Tag1|Channel1.Device1.Tag2>

WebSocket JSON:

```

{"command": "subscribe", "server": "opcserverim.Instance.1", "group": "MyGroup1",
"tag": "Channel1.Device1.Tag1|Channel1.Device1.Tag2", "poll_rate": 30000}
{"command": "subscribe", "server": "opcserverim.Instance.1", "group": "MyGroup1",
"tag": "Channel1.Device1.Tag1\u000DChannel1.Device1.Tag2"}

```

:

```

{"command": "subscribe", "success": true, "processed": 1}

```

```
{"command": "unsubscribe", "success": false, "processed": 0}
```

## 6.8

```

HTTP-
OPC-
: unsubscribe.json
:
server (
,
) - OPC-
host (
,
) - IP- OPC-
group (
,
) -
tag,
group «*»,
tag (
,
) - (
group.
)

```

HTTP:

```

http://127.0.0.1/unsubscribe.json?server=opcserver Instance.1&tag=Channel1.Device1.Tag1|
Channel1.Device1.Tag2
http://127.0.0.1/unsubscribe.json?server=opcserver Instance.1&group=MyGroup
http://127.0.0.1/unsubscribe.json?server=opcserver Instance.1&group=*

```

WebSocket JSON:

```

{"command": "unsubscribe", "server": "opcserver Instance.1", "tag":
"Channel1.Device1.Tag1| Channel1.Device1.Tag2"}
{"command": "unsubscribe", "server": "opcserver Instance.1", "group": "MyGroup"}
{"command": "unsubscribe", "server": "opcserver Instance.1", "group": "*"}
:
{"command": "unsubscribe", "success": true, "processed": 1}

{"command": "unsubscribe", "success": false, "processed": 0}

```

## 6.9

```

: list-subscribed.json
:
server ( , ) - OPC- .
host ( , ) - IP- OPC- .
group ( , ) -
group «*»,

```

```

HTTP:
http://127.0.0.1/list-subscribed.json?server=opcserverim.Instance.1&group=*

```

```

WebSocket JSON:
{"command": "list-subscribed", "server": "opcserverim.Instance.1", "group": "*" }
:
{"command": "list-subscribed", "success": true, "data":
[{"name": "Channell.Devicel.Tag1", "group": "MyGroup"},
{"name": "Channell.Devicel.Tag2", "group": "MyGroup"},
{"name": "Channell.Devicel.Tag3", "group": "MyGroup"}]}

```

## 6.10

```

OPC , OPC HTTP- OPC- .
:
(SyncRead). OPC
: read.json
:
server ( , ) - OPC- .
host ( , ) - IP- OPC- .

```

```

group (
    ,
    tag,
    group «*»,
)

tag (
    ,
) - (
    group.
).

```

HTTP:

http://127.0.0.1/read.json?server=opcserver1sim.Instance.1&tag=Channel1.Device1.Tag1

WebSocket JSON:

```

{"command": "read", "server": "opcserver1sim.Instance.1", "tag":
"Channel1.Device1.Tag1"}

```

:

```

{"command": "read", "success": true, "processed": 1,
"data": [{"name": "Channel1.Device1.Tag1", "group": "MyGroup",
"value": "123", "quality": "0xC0", "timestamp": "2016-10-27 06:23:24", "data_type":
8}]}

```

WebSocket

:

```

{"command": "read", "async": true, "success": true, "processed": 1,
"data": [{"name": "Channel1.Device1.Tag1", "group": "MyGroup",
"value": "123", "quality": "0xC0", "timestamp": "2016-10-27 06:23:24", "data_type":
8}]}

```

data -

:

name -

group -

value -

data\_type - \_\_\_\_\_

quality - \_\_\_\_\_ OPC , OPC

timestamp - \_\_\_\_\_ , OPC

## 6.11

```

        OPC
    ,
    OPC-
    .

: write.json

:

server (
    ,
) - OPC-
.

host (
    ,
) - IP- OPC-
.

group (
    ,
) -
.

tag (
    ,
) - OPC.

value (
    ,
) -
.

data_type (
    ,
) -
.

```

HTTP:

```

http://127.0.0.1/write.json?
server=opcserver Instance.1&group=MyGroup&tag=Channel1.Device1.Tag1&value=321&data_type=8

```

WebSocket JSON:

```

{"command": "write", "server": "opcserver Instance.1", "group": "MyGroup", "tag":
"Channel1.Device1.Tag1", "value": "321", "data_type": 8}

```

:

```

{"command": "write", "success": true, "processed": 1}

```

## 6.12

```

-
,
jQuery
demo.html
«files»

URL-
:

HTTP : http://127.0.0.1:8080/demo.html

```

HTTPS : <https://127.0.0.1:8443/demo.html> ( ,  
SSL ,

).  
HTTP UA : [http://127.0.0.1:8080/demo\\_http\\_ua.html](http://127.0.0.1:8080/demo_http_ua.html)

WebSocket : [http://127.0.0.1:8080/demo\\_websocket.html](http://127.0.0.1:8080/demo_websocket.html)

,  
OPC-HTTP Gateway. ,  
.