SonicWall® SonicOS API 6.5.1
Reference
Schema Structure ........................................................................................................... 25
Object: Address Object ..................................................................................................... 25
Collection: Address Objects ............................................................................................. 26
Schema Attributes ............................................................................................................. 26

API: Address Objects – MAC .......................................................................................... 28
Endpoint ............................................................................................................................. 28
Schema Structure ............................................................................................................. 28
Object: Address Object ..................................................................................................... 28
Collection: Address Object ............................................................................................. 29
Schema Attributes ............................................................................................................. 29

API: Address Objects – FQDN ....................................................................................... 31
Endpoint ............................................................................................................................. 31
Schema Structure ............................................................................................................. 31
Object: Address Object ..................................................................................................... 31
Collection: Address Object ............................................................................................. 32
Schema Attributes ............................................................................................................. 32

API: Address Groups — IPv4 .......................................................................................... 34
Endpoint ............................................................................................................................. 34
Schema Structure ............................................................................................................. 34
Object: Address Group ...................................................................................................... 34
Collection: Address Group ............................................................................................. 35
Schema Attributes ............................................................................................................. 35

API: Address Groups — IPv6 .......................................................................................... 38
Endpoint ............................................................................................................................. 38
Schema Structure ............................................................................................................. 38
Object: Address Group ...................................................................................................... 38
Collection: Address Group ............................................................................................. 39
Schema Attributes ............................................................................................................. 39

API: Schedule Objects ................................................................................................... 42
Endpoint ............................................................................................................................. 42
Schema Structure ............................................................................................................. 42
Object: Schedule ............................................................................................................... 42
Collection: Schedule ........................................................................................................ 43
Schema Attributes ............................................................................................................. 43

API: Service Objects ...................................................................................................... 48
Endpoint ............................................................................................................................. 48
Schema Structure ............................................................................................................. 48
Object: Service Object ...................................................................................................... 48
Collection: Service Object ............................................................................................. 49
Schema Attributes ............................................................................................................. 49

API: Service Groups ....................................................................................................... 52
Endpoint ............................................................................................................................. 52
<table>
<thead>
<tr>
<th>API: Access Rules – IPv4</th>
<th>107</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schema Attributes</td>
<td>107</td>
</tr>
<tr>
<td><strong>API: Access Rules – IPv6</strong></td>
<td>116</td>
</tr>
<tr>
<td>Endpoint</td>
<td>116</td>
</tr>
<tr>
<td>Schema Structure</td>
<td>116</td>
</tr>
<tr>
<td>Collection: Access Rules – IPv6</td>
<td>118</td>
</tr>
<tr>
<td>Schema Attributes</td>
<td>118</td>
</tr>
<tr>
<td><strong>API: Route Policies – IPv4</strong></td>
<td>127</td>
</tr>
<tr>
<td>Endpoint</td>
<td>127</td>
</tr>
<tr>
<td>Schema Structure</td>
<td>127</td>
</tr>
<tr>
<td>Object: Route Policies – IPv4</td>
<td>127</td>
</tr>
<tr>
<td>Collection: Route Policies – IPv4</td>
<td>128</td>
</tr>
<tr>
<td>Schema Attributes</td>
<td>128</td>
</tr>
<tr>
<td><strong>API: Route Policies – IPv6</strong></td>
<td>132</td>
</tr>
<tr>
<td>Endpoint</td>
<td>132</td>
</tr>
<tr>
<td>Schema Structure</td>
<td>132</td>
</tr>
<tr>
<td>Object: Route Policies – IPv6</td>
<td>132</td>
</tr>
<tr>
<td>Collection: Route Policies – IPv6</td>
<td>133</td>
</tr>
<tr>
<td>Schema Attributes</td>
<td>133</td>
</tr>
<tr>
<td><strong>SonicWall Support</strong></td>
<td>137</td>
</tr>
<tr>
<td>About This Document</td>
<td>138</td>
</tr>
</tbody>
</table>
About SonicOS API

- **SonicOS API Function** on page 5
  - Enabling through the Management Interface on page 5
  - Enabling through the CLI on page 6
- **Supported Request Methods** on page 6
  - Supported HTTP Headers on page 6
  - HTTP Status Codes on page 9
  - Status and Error Representation on page 9
- **Client Authentication** on page 11
  - HTTP Basic Authentication on page 12
- **Examples** on page 13

**SonicOS API Function**

SonicOS API provides an alternative to the SonicOS Command Line Interface (CLI) for configuring selected functions.

SonicOS API is disabled by default in SonicOS. Any attempts to access SonicOS API while it is disabled results in an HTTP 403 Forbidden error. To use the SonicOS API, you must enable it, either through the SonicOS Management Interface or from the CLI.

**Topics:**

- Enabling through the Management Interface on page 5
- Enabling through the CLI on page 6

**Enabling through the Management Interface**

*To enable SonicOS API through the management interface:*

1. Navigate to **MANAGE | Network > Appliance | Base Settings**.
2. Scroll to the **SonicOS API** section.
3. Select **Enable SonicOS API**.
4. Click **Accept**.
Enabling through the CLI

Starting at the `config#` prompt:

```
config(<serial number>)# administration
(config-administration)# sonicos-api
(config-administration)# commit
```

Supported Request Methods

SonicOS API utilizes four of the methods defined in the HTTP protocol (RFC 7231 and RFC 5789) to create, read, update and delete (CRUD) resources. Supported HTTP request methods describes the supported HTTP methods.

<table>
<thead>
<tr>
<th>HTTP method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>Retrieves the specified resource or collection of resources. GET is a read-only operation that does not alter appliance state or configuration. A GET operation should not contain a request-body.</td>
</tr>
<tr>
<td>POST</td>
<td>Submits data to be processed by the specified resource or collection of resources. In most cases, the POST verb is used by SonicOS APIs to create and add a resource to a collection of resources (for example, add a new MAC address-object to collection of objects).</td>
</tr>
<tr>
<td>PUT</td>
<td>Updates the specified resource. The data included in the PUT request-body replaces the previous configuration.</td>
</tr>
<tr>
<td>DELETE</td>
<td>Deletes the specified resource or collection of resources.</td>
</tr>
</tbody>
</table>

**Supported HTTP header request and response formats**

<table>
<thead>
<tr>
<th>Type</th>
<th>Example</th>
</tr>
</thead>
</table>
| Text/plain     | GET /api/sonicos/address-objects/mac
Accept: text/plain |
| Application/JSON | POST /api/sonicos/address-objects/mac
Content-type: application/json
Accept: application/json |

```
{
    "address_object": {
        "mac": {
            "name": "001122334455"
        }
    }
}
```

**Supported HTTP Headers**

- **Content-type** Specifies the format (MIME type) of the request body (input).
- **Accept** Specifies the format of the response body (output).
Supported HTTP MIME Types

SonicOS supports these HTTP MIME types:

- Text/plain
- Application/JSON

These HTTP headers define the request and response format:

- **Content-type** – Specifies the format (MIME type) of the request body (input)
- **Accept** – Specifies the format of the response body (output)

NOTE: The headers can be used to obtain mixed input/output. See examples below for reference.

Examples

Topics:

- Application/JSON on page 7
- Text/Plain on page 8

Application/JSON

When specified, the request and/or response body is expected to be in SonicOS API JSON format.

Request

```plaintext
POST /api/sonicos/address-objects/mac
Content-type: application/json
Accept: application/json

{
   "address_object": {
      "mac": {
         "name": "001122334455",
         "address": "001122334455",
         "multi_homed": true,
         "zone": "LAN"
      }
   }
}
```

Response

```plaintext
HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8

{
   "status": {
      "success": true,
      "cli": {
         "depth": 1,
         "mode": "config_mode",
         "configuring": true
      }
   }
}
```
When specified, the request and/or response body is expected to be in SonicOS CLI plain-text command format.

**Topics:**
- Request 1 on page 8
- Request 2 on page 8

**Request 1**

GET /api/sonicos/address-objects/mac
Accept: text/plain

**Response**

HTTP/1.0 200 OK
Server: SonicWALL
Content-type: text/plain; charset=UTF-8

address-object mac example address 001122334455
zone LAN
multi-homed
exit

**Request 2**

POST /api/sonicos/direct/cli
Content-type: text/plain
Accept: application/json

address-object mac example address 001122334455
zone LAN
multi-homed
exit

**Response**

HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8

```json
{
  "status": {
    "success": true,
    "cli": {
      "depth": 1,
      "mode": "config_mode",
      "configuring": true,
      "pending_config": true
    }
  }
}
```
Status and Error Representation

All plain text output from the last backend CLI command executed is captured and returned back to the client. If the command executed was not a show command and the requested operation succeeded, then the response body is empty. This is consistent with the CLI when executing a command via SSH or the serial console in that status is only rendered to the console upon error.

A JSON status object is guaranteed to be returned in the response body when performing a POST, PUT, or DELETE operation or upon error(s) encountered when processing a request.

Topics:
- HTTP Status Codes on page 9
- Application/JSON on page 10

HTTP Status Codes

SonicOS API uses standard HTTP status codes to report success or failure when servicing a request.

<table>
<thead>
<tr>
<th>Code</th>
<th>Status Text</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>OK</td>
<td>The request succeeded.</td>
</tr>
<tr>
<td>400</td>
<td>Bad Request</td>
<td>An invalid request was submitted. Verify that the request URI is correct and that the request body is as expected.</td>
</tr>
<tr>
<td>401</td>
<td>Not Authorized</td>
<td>The user is unauthenticated or lacks the required privileges for the operation requested.</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden</td>
<td>The request was understood by the server but denied. The response body notes the reason why the request was denied.</td>
</tr>
<tr>
<td>404</td>
<td>Not Found</td>
<td>The resource specified was not found.</td>
</tr>
<tr>
<td>405</td>
<td>Method Not Allowed</td>
<td>The HTTP verb specified is not allowed or supported by the resource specified.</td>
</tr>
<tr>
<td>406</td>
<td>Not Acceptable</td>
<td>The MIME type specified in the HTTP Content-type and/or Accept header is not supported.</td>
</tr>
<tr>
<td>413</td>
<td>Request body too large</td>
<td>Maximum size of the request body was exceeded.</td>
</tr>
<tr>
<td>414</td>
<td>Request URL too long</td>
<td>The request URL exceeded the maximum size allowed or contains extra/unknown parameters (directories).</td>
</tr>
<tr>
<td>500</td>
<td>Internal Server Error</td>
<td>The request failed due to an internal server error. The response body should note the reason why the request failed.</td>
</tr>
<tr>
<td>503</td>
<td>No resources</td>
<td>Maximum number of sessions was exceeded.</td>
</tr>
</tbody>
</table>
Application/JSON

A JSON status object is guaranteed to be returned in the response body when performing a POST, PUT, or DELETE operation or upon error(s) encountered when processing a request.

Topics:
- [Schema Structure](#) on page 10
- [Schema Attributes](#) on page 10

### Schema Structure

```json
{
  "status": {
    "success": {boolean},
    "cli": {
      "depth": {number},
      "mode": "{string}\n      "command": "{string}\n      "configuring": {boolean},
      "pending_config": {boolean},
      "restart_required": "{string}\n    }
    "info": [{
      "level": "{string}\n      "code": "{string}\n      "message": "{string}\n    } ...
  }
}
```

### Schema Attributes

**Schema attributes**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>object</td>
<td>Status object.</td>
</tr>
<tr>
<td>status.success</td>
<td>boolean (true</td>
<td>false)</td>
</tr>
<tr>
<td>status.cli</td>
<td>object</td>
<td>CLI status. <strong>NOTE:</strong> This attribute is included only when an API sent one or more commands to the CLI backend.</td>
</tr>
</tbody>
</table>
| status.cli.depth| number (uint8)  | Current mode depth of the CLI:  
|                 |                 | • 0 = top-level mode  
|                 |                 | • >= 1 config mode |
| status.cli.mode | string          | Name of the current mode.                                                   |
| status.cli.command | string    | Command last executed. **NOTE:** This attribute is only included upon command error(s). |
| status.cli.configuring | boolean (true|false) | Boolean configuring flag. Should always be true upon one or more consecutive POST, PUT or DELETE API calls that modify the configuration. |
### Client Authentication

SonicOS API currently offers two mechanisms for client authentication:

- HTTP Basic Authentication (RFC 2617)
- Challenge-Handshake Authentication (CHAP)

Regardless of the authentication mechanism used, only:

- A single administrator can manage (modify configuration) at any given time. This remains true regardless of where an admin logged in (web management UI, CLI, GMS, or SonicOS API).
- Users with full admin privileges are allowed to access SonicOS API.
- A single SonicOS API session is currently allowed.

### Endpoint

Both authentication mechanisms share the same endpoint for client login and logout.

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>GET</th>
<th>POST</th>
<th>PUT</th>
<th>DELETE</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/auth</td>
<td>Empty</td>
<td>Empty</td>
<td>—</td>
<td>Empty</td>
</tr>
</tbody>
</table>
HTTP Basic Authentication

HTTP Basic Authentication is the simplest method for client authentication as it does not require cookies, session identifiers, etc. HTTP Basic Authentication uses the standard Authentication HTTP header to pass user credentials between the client and server. Because HTTP Basic Authentication provides no means for protecting the confidentiality of a user’s credentials, SonicOS API requires user credentials to be transmitted over HTTPS.

For SonicOS API HTTP Basic Authentication, use the Linux command-line curl command with the -u option:

- Login:
  
curl -k -i -u admin:password -X POST https://a.b.c.d/api/sonicos/auth
- Logout:
  
curl -k -i -X DELETE https://a.b.c.d/api/sonicos/auth

Challenge-Handshake Authentication (CHAP)

In addition to HTTP Basic Authentication, SonicOS API supports the same CHAP-style authentication mechanism used by both the SonicOS Management Interface GUI and GMS for client authentication.

Clients must first perform a CHAP challenge initiate request by invoking a call to GET /api/sonicos/auth:

HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8

{
    "id": "{string}",
    "challenge": "{string}"
}

**id**: Type: string (hexadecimal number)
Description: CHAP ID
Example: 0b

**challenge**: Type: string (hexadecimal #)
Description: Hexadecimal formatted randomly generated number
Example: EA7F57F37595B6891C222EF284C05D84

Clients must then generate a one-way hash (CHAP digest) using the user’s credentials and the parameters returned via the initiate request. Please see ‘auth.js’ in the SonicOS Web UI code for a working reference on how to calculate the digest.

When the CHAP digest is generated, it is then packaged up via a JSON formatted request to POST /api/sonicos/auth:

{
    "override": {boolean},
    "id": "{string}",
    "user": "{string}",
    "digest": "{string}"
Examples

Topics:

- Example - Commit Pending Configuration on page 13
- Example - Address Object API Calls on page 15

Example - Commit Pending Configuration

All SonicOS APIs that modify configuration (POST, PUT, DELETE) do not take effect immediately. Rather, configuration is staged and is not pushed to run-time config and saved to flash/permanent storage until API clients explicitly execute a POST request to /api/sonicos/config/pending. This is the same behavior as in the SonicOS CLI and equivalent to invoking the commit command from the top-level config mode.

Pending configuration can be canceled (deleted) at any time by executing a DELETE request to /api/sonicos/config/pending. Any/all pending configuration is canceled upon client session termination, whether due to idle-timeout or explicit logout. In this case, all unsaved changes are lost. It is the client’s responsibility to either commit pending configuration after each POST/PUT/DELETE API call or maintain pending changes on the client side to be restored in a later session.

Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/config/pending</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: N/A</td>
<td>Empty</td>
</tr>
</tbody>
</table>

Topics:

- Schema on page 14
• Examples on page 14

Schema

Schema Structure
A schema is not really applicable here as POST, PUT and DELETE HTTP body is expected to be empty. However, GET returns any/all pending (unsaved) configuration.

Schema Attributes
Not applicable.

Examples

Topics:
• # GET Pending Changes (unchanged) on page 14
• # GET Pending Changes on page 14
• # POST Pending Changes on page 15

# GET Pending Changes (unchanged)

Request:
GET /api/sonicos/config/pending
Accept: application/json

Response:
HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8
{
}

# GET Pending Changes

Request:
GET /api/sonicos/config/pending
Accept: application/json

Response:
HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8
{
    "address_objects": [
    
    
    ]
}
# POST Pending Changes

**Request:**

POST /api/sonicos/config/pending
Accept: application/json

**Response:**

HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8
{
  "status": {
    "success": true,
    "cli": {
      "depth": 1,
      "mode": "config_mode",
      "configuring": true,
      "pending_config": false,
      "restart_required": "NONE"
    },
    "info": [
      { "level": "info", "code": "E_OK", "message": "Success." }
    ]
  }
}

---

**Example - Address Object API Calls**

**Topics:**

- # Create a new IPv4 Address Object named Web Server on page 15
- # Modify the Web Server Address Object host IP on page 16
- # Delete the Web Server Address Object on page 16

**# Create a new IPv4 Address Object named Web Server**

POST /api/sonicos/address-objects/ipv4
Content-type: application/json
{
  "address_object": {

```json
  
  ,"ipv4": {
    "name": "B",
    "host": {
      "ip": "2.2.2.2"
    },
    "zone": "WAN"
  }
  
```
}

---


"ipv4": {
   "name": "Web Server",
   "zone": "DMZ",
   "host": {
      "ip": "192.168.168.168"
   }
},

## Modify the Web Server Address Object host IP

PUT /api/sonicos/address-objects/ipv4/name/Web%20Server
Content-type: application/json

```
{
   "address_object": {
      "ipv4": {
         "host": {
            "ip": "192.168.168.1"
         }
      }
   }
}
```

## Delete the Web Server Address Object

DELETE /api/sonicos/address-objects/ipv4/name/Web%20Server
API: Config - Pending

- About Modifying Configuration API on page 17
- Endpoint on page 17
- Schema Structure on page 17
- Examples on page 18

About Modifying Configuration API

All SonicOS APIs that modify configuration (POST, PUT, DELETE) do not take effect immediately. Rather, configuration is staged and will not be pushed to run-time config and saved to flash/permanent storage until API clients explicitly execute a POST request to /api/sonicos/config/pending. This is the same behavior as SonicOS CLI and equivalent to invoking the commit command from the top-level config mode.

Pending configuration can be canceled (deleted) at any time by executing a DELETE request to /api/sonicos/config/pending. It should be noted that any/all pending configuration is canceled (deleted) upon client session termination, whether due to idle-timeout or explicit logout. In this case, all unsaved changes will be lost so it is the client’s responsibility to either commit pending configuration after each POST/PUT/DELETE API call or maintain pending changes on the client side to be restored in a later session.

Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>Empty</td>
</tr>
<tr>
<td>POST</td>
<td>Empty</td>
</tr>
<tr>
<td>PUT</td>
<td>—</td>
</tr>
<tr>
<td>DELETE</td>
<td>Empty</td>
</tr>
</tbody>
</table>

| URI: /api/sonicos/config/pending | Schema: N/A |

Schema Structure

A schema is not really applicable here as POST, PUT and DELETE HTTP body is expected to be empty. However, GET will return any/all pending (unsaved) configuration so please see all schemas in the following chapters.

Schema Attributes

Not applicable.
Examples

Topics:
- GET Pending Changes (Unchanged) on page 18
- GET Pending Changes on page 18
- GET Pending Changes on page 19
- POST Pending Changes on page 19

GET Pending Changes (Unchanged)

Request:
GET /api/sonicos/config/pending
Accept: application/json

Response:
HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8

```
{}
```
GET Pending Changes

Request:
GET /api/sonicos/config/pending
Accept: application/json

Response:
HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8

{
    "address_objects": [
        {
            "pending": "ADD",
            "ipv4": {
                "name": "B",
                "host": {
                    "ip": "2.2.2.2"
                }
            },
            "zone": "WAN"
        }
    ]
}

POST Pending Changes

Request:
POST /api/sonicos/config/pending
Accept: application/json

Response:
HTTP/1.0 200 OK
Server: SonicWALL
Content-type: application/json; charset=UTF-8

{
    "status": {
        "success": true,
        "cli": {
            "depth": 1,
            "mode": "config_mode",
            "configuring": true,
            "pending_config": false,
            "restart_required": "NONE"
        }
    },
    "info": [ {
        "level": "info", "code": "E_OK", "message": "Success." }
    ]
}

API: Restart

- About Restarting API on page 20
- Endpoint on page 20
- Schema Structure on page 20
- Example on page 20

About Restarting API

Restarts SonicOS (and chassis) immediately or after an interval of time.

Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/restart [ / chassis ] [ /at/{YYYYMMDDHHMMSS} ] [ /in/{UINT32} { /minutes</td>
<td>/hours</td>
</tr>
</tbody>
</table>

Schema: N/A

Schema Structure

Not applicable.

Schema Attributes

Not applicable.

Example

POST /api/sonicos/restart
POST /api/sonicos/restart/now
POST /api/sonicos/restart/chassis/now
POST /api/sonicos/restart/in/3/days
API: Address Objects – IPv4

- **Endpoint** on page 22
- **Schema Structure** on page 22

### Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/address-objects/ipv4</td>
<td>GET Empty, POST Required, PUT Required, DELETE Required</td>
</tr>
<tr>
<td>Schema: collection#address-object-ipv4-config</td>
<td></td>
</tr>
</tbody>
</table>

| URI: /api/sonicos/address-objects/ipv4/name/{NAME} | GET Empty, — Required, PUT Ignored |
| Schema: object#address-object-ipv4-config | |

| URI: /api/sonicos/address-objects/ipv4/uuid/{UUID} | GET Empty, — Required, PUT Ignored |
| Schema: object#address-object-ipv4-config | |

### Schema Structure

**Topics:**
- **Object: Address Object** on page 22
- **Collection: Address Object** on page 23
- **Schema Attributes** on page 23

### Object: Address Object

```json
{
    "address_object": {
        "ipv4": {
            "name": "{string}",
            "uuid": "{string}",
            "host": {
                "ip": "{string}" |
                "range": {
                    "begin": "{string}",
                    "end": "{string}" |
                },
                "network": {
                    "subnet": "{string}" |
                }
            }
        }
    }
}
```
"mask": "string"  
},  
"zone": "string"  
}  
}

### Collection: Address Object

```
{
   "address_objects": [  
      object#address-object-ipv4-config,  
      ...
   ]
}
```

### Schema Attributes

**address_object:**

- **Type:** object  
- **Flags:** -none-  
- **Description:** Add/edit address object.

**address_objects:**

- **Type:** array  
- **Flags:** -none-  
- **Description:** Address object collection.

**address_object.ipv4:**

- **Type:** object  
- **Flags:** key  
- **Description:** IPV4 address object.

**address_object.ipv4.name:**

- **Type:** string  
- **Flags:** key  
- **Description:** Host/network/range address object name.

**address_object.ipv4.uuid:**

- **Type:** string  
- **Flags:** key  
- **Description:** UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

**address_object.ipv4.host:**

- **Type:** object  
- **Flags:** -none-  
- **Description:** Address object host.
address_object.ipv4.host.ip:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 host address in the form: D.D.D.D.

address_object.ipv4.range:
  Type: object
  Flags: -none-
  Description: Address object range.

address_object.ipv4.range.begin:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 starting range in the form: D.D.D.D.

address_object.ipv4.range.end:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 ending range in the form: D.D.D.D.

address_object.ipv4.network:
  Type: object
  Flags: -none-
  Description: Address object network.

address_object.ipv4.network.subnet:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 network in the form: D.D.D.D.

address_object.ipv4.network.mask:
  Type: string (subnet)
  Flags: -none-
  Description: IPv4 netmask in decimal dotted or CIDR form: D.D.D.D OR /D

address_object.ipv4.zone:
  Type: string
  Flags: -none-
  Description: Zone object name.
API: Address Objects – IPv6

- Endpoint on page 25
- Schema Structure on page 25

 Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
<th>GET</th>
<th>POST</th>
<th>PUT</th>
<th>DELETE</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/address-objects/ipv6</td>
<td>Empty</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Schema: collection#address-object-ipv6-config</td>
<td>Empty</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/address-objects/ipv6/name/{NAME}</td>
<td>Empty</td>
<td>Empty</td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>Schema: object#address-object-ipv6-config</td>
<td>Empty</td>
<td></td>
<td></td>
<td></td>
<td>Ignored</td>
</tr>
<tr>
<td>URI: /api/sonicos/address-objects/ipv6/uuid/{UUID}</td>
<td>Empty</td>
<td>Empty</td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>Schema: object#address-object-ipv6-config</td>
<td>Empty</td>
<td></td>
<td></td>
<td></td>
<td>Ignored</td>
</tr>
</tbody>
</table>

 Schema Structure

Topics:
- Object: Address Object on page 25
- Collection: Address Objects on page 26
- Schema Attributes on page 26

 Object: Address Object

```json
{
  "address_object": {
    "ipv6": {
      "name": "{string}",
      "uuid": "{string}",
      "host": {
        "ip": "{string}",
      },
      | "range": {
        "begin": "{string}",
        "end": "{string}",
      },
      | "network": {
        "subnet": "{string}",
      }
    }
  }
```
Collection: Address Objects

{
    "address_objects": [
        object#address-object-ipv6-config,
        ...
    ]
}

Schema Attributes

address_object:
    Type: object
    Flags: -none-
    Description: Add/edit address object.

address_objects:
    Type: array
    Flags: -none-
    Description: Address object collection.

address_object.ipv6:
    Type: object
    Flags: key
    Description: IPV6 address object.

address_object.ipv6.name:
    Type: string
    Flags: key
    Description: Host/network/range address object name.

address_object.ipv6.uuid:
    Type: string
    Flags: key
    Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

address_object.ipv6.host:
    Type: object
    Flags: -none-
    Description: Address object host.
address_object.ipv6.host.ip:
  Type: string (ip)
  Flags: -none-
  Description: IPv6 host address in the form: HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH

address_object.ipv6.range:
  Type: object
  Flags: -none-
  Description: Address object range.

address_object.ipv6.range.begin:
  Type: string (ip)
  Flags: -none-
  Description: IPv6 starting range in the form: HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH

address_object.ipv6.range.end:
  Type: string (ip)
  Flags: -none-
  Description: IIPv6 ending range in the form: HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH

address_object.ipv6.network:
  Type: object
  Flags: -none-
  Description: Address object network.

address_object.ipv6.network.subnet:
  Type: string (ip)
  Flags: -none-

address_object.ipv6.network.mask:
  Type: string (v6 prefix)
  Flags: -none-
  Description: Network prefix.

address_object.ipv6.zone:
  Type: string
  Flags: -none-
  Description: Zone object name.
API: Address Objects – MAC

- **Endpoint** on page 28
- **Schema Structure** on page 28

**Endpoint**

<table>
<thead>
<tr>
<th>URI: /api/sonicos/address-objects/mac</th>
<th>Schema: collection#address-object-mac-config</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET Empty POST Required PUT Required</td>
<td>DELETE Required</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>URI: /api/sonicos/address-objects/mac/name/{NAME}</th>
<th>Schema: object#address-object-mac-config</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET Empty — PUT Required</td>
<td>DELETE Ignored</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>URI: /api/sonicos/address-objects/mac/uuid/{UUID}</th>
<th>Schema: object#address-object-mac-config</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET Empty — PUT Required</td>
<td>DELETE Ignored</td>
</tr>
</tbody>
</table>

**Schema Structure**

**Topics:**
- **Object:** Address Object on page 28
- **Collection:** Address Object on page 29
- **Schema Attributes** on page 29

**Object: Address Object**

```json
{
    "address_object": {
        "mac": {
            "name": "[string]",
            "uuid": "[string]",
            "address": "[string]",
            "zone": "[string]",
            "multi_homed": [boolean]
        }
    }
}
```
Collection: Address Object

```json
{
    "address_objects": [
        object#address-object-mac-config,
        ...
    ]
}
```

**Schema Attributes**

**address_object:**
- Type: object
- Flags: -none-
- Description: address object.

**address_objects:**
- Type: array
- Flags: -none-
- Description: Address object collection.

**address_object.mac:**
- Type: object
- Flags: key
- Description: MAC address object.

**address_object.mac.name:**
- Type: string
- Flags: key
- Description: MAC address object name.

**address_object.mac.uuid:**
- Type: string
- Flags: key
- Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHH

**address_object.mac.address:**
- Type: string (mac)
- Flags: -none-
- Description: Address object MAC address in the form: HH:HH:HH:HH:HH:HH or HHHHHHHHHHHHT or HH-HH-HH-HH-HH.

**address_object.mac.zone:**
- Type: string
- Flags: -none-
- Description: Zone object name.
**address_object.mac.multi_homed:**

- **Type:** boolean (true/false)
- **Flags:** -none-
- **Description:** Enable multi-homed host.
API: Address Objects – FQDN

- **Endpoint** on page 31
- **Schema Structure** on page 31

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicoss/address-objects/fqdn</td>
<td>GET Empty, POST Required, PUT Required, DELETE Required</td>
</tr>
<tr>
<td>Schema: collection#address-object-fqdn-config</td>
<td></td>
</tr>
</tbody>
</table>

| URI: /api/sonicoss/address-objects/fqdn/name/{NAME} | GET Empty, PUT Required, DELETE Ignored |
| Schema: object#address-object-fqdn-config |

| URI: /api/sonicoss/address-objects/fqdn/uuid/{UUID} | GET Empty, PUT Required, DELETE Ignored |
| Schema: object#address-object-fqdn-config |

## Schema Structure

**Topics:**
- **Object: Address Object** on page 31
- **Collection: Address Object** on page 32
- **Schema Attributes** on page 32

### Object: Address Object

```json
{
    "address_object": {
        "fqdn": {
            "name": "[string]",
            "uuid": "[string]",
            "domain": "[string]",
            "zone": "[string]",
            "dns_ttl": [number]
        }
    }
}
```
Collection: Address Object

```
{
    "address_objects": [
        object#address-object-fqdn-config,
        ...
    ]
}
```

Schema Attributes

**address_object:**
- Type: object
- Flags: -none-
- Description: address object.

**address_objects:**
- Type: array
- Flags: -none-
- Description: Address object collection.

**address_object.fqdn:**
- Type: object
- Flags: key
- Description: fqdn address object.

**address_object.fqdn.name:**
- Type: string
- Flags: key
- Description: FQDN address object name.

**address_object.fqdn.uuid:**
- Type: string
- Flags: key
- Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

**address_object.fqdn.domain**
- Type: string (fqdn)
- Flags: -none-
- Description: FQDN in the form: example.com or *.example.com.

**address_object.fqdn.zone:**
- Type: string
- Flags: -none-
- Description: Zone object name.

**address_object.fqdn.dns_ttl**
- Type: number (uint16))
Flags: -none-
Description: Integer in the form: D OR 0xHHHH
API: Address Groups — IPv4

- **Endpoint** on page 34
- **Schema Structure** on page 34

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>GET</th>
<th>POST</th>
<th>PUT</th>
<th>DELETE</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/address-groups/ipv4</td>
<td>Empty</td>
<td>Required</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>Schema: collection#address-group-ipv4-config</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/address-groups/ipv4/name/{NAME}</td>
<td>Empty</td>
<td>—</td>
<td>Required</td>
<td>If deleting member(s)</td>
</tr>
<tr>
<td>Schema: object#address-group-ipv4-config</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/address-groups/ipv4/uuid/{UUID}</td>
<td>Empty</td>
<td>—</td>
<td>Required</td>
<td>If deleting member(s)</td>
</tr>
<tr>
<td>Schema: object#address-group-ipv4-config</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Schema Structure

**Topics:**
- **Object: Address Group** on page 34
- **Collection: Address Group** on page 35
- **Schema Attributes** on page 35

### Object: Address Group

```
{
    "address_group": {
        "ipv4": {
            "name": "{string}",
            "uuid": "{string}",
            "address_group": {
                "ipv4": [
                    {
                        "name": "{string}"
                    },
                    ...,
                ],
            },
        },
    },
}
```
"address_object": {
  "ipv4": [
    {
      "name": "{string}"
    },
    ...,
  ],
  "mac": [
    {
      "name": "{string}"
    },
    ...,
  ],
  "fqdn": [
    {
      "name": "{string}"
    },
    ...
  ]
}
}
}

Collection: Address Group
{
  "address_objects": [
    object@address-group-ipv4-config,
    ...,
  ]
}

Schema Attributes

address_group:
  Type: object
  Flags: -none-
  Description: Address group.

address_groups:
  Type: array
  Flags: -none-
  Description: Address group collection.

address_group.ipv4:
  Type: object
  Flags: key
  Description: ipv4 address group.
address_group.ipv4.name:
  Type: string
  Flags: key
  Description: IPv4 address group name.

address_group.ipv4.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHHH

address_group.ipv4.address_group:
  Type: object
  Flags: -none-
  Description: Assign address group to group.

address_group.ipv4.address_group.ipv4:
  Type: array
  Flags: -none-
  Description: IPV4 address group.

address_group.ipv4.address_group.ipv4.name:
  Type: string
  Flags: -none-
  Description: Group address object name.

address_group.ipv4.address_object:
  Type: object
  Flags: -none-
  Description: Assign an FQDN address object to group.

address_group.ipv4.address_object.ipv4:
  Type: array
  Flags: -none-
  Description: IPV4 address object.

address_group.ipv4.address_object.ipv4.name:
  Type: string
  Flags: -none-
  Description: Host/network/range address object name.

address_group.ipv4.address_object.mac:
  Type: array
  Flags: -none-
  Description: MAC address object.

address_group.ipv4.address_object.mac.name:
  Type: string
  Flags: -none-
  Description: MAC address object name.
address_group.ipv4.address_object.fqdn:
  - Type: array
  - Flags: -none-
  - Description: FQDN address object.

address_group.ipv4.address_object.fqdn.name:
  - Type: string
  - Flags: -none-
  - Description: FQDN address object name.
API: Address Groups — IPv6

- **Endpoint** on page 38
- **Schema Structure** on page 38

### Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/address-groups/ipv6</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#address-group-ipv6-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/address-groups/ipv6/name/{NAME}</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: object#address-group-ipv6-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/address-groups/ipv6/uuid/{UUID}</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: object#address-group-ipv6-config</td>
<td></td>
</tr>
</tbody>
</table>

### Schema Structure

**Topics:**
- **Object: Address Group** on page 38
- **Collection: Address Group** on page 39
- **Schema Attributes** on page 39

### Object: Address Group

```
{
    "ipv6": {
        "name": "{string}",
        "uuid": "{string}",

        "address_group": {
            "ipv4": [
                { "name": "{string}" },
                ...
            ],

            "ipv6": [
```
{
    "name": "[string]"
},
...
}

"address_object": {
    "ipv4": [
        {
            "name": "[string]"
        },
        ...
    ],
    "ipv6": [
        {
            "name": "[string]"
        },
        ...
    ],
    "mac": [
        {
            "name": "[string]"
        },
        ...
    ],
    "fqdn": [
        {
            "name": "[string]"
        },
        ...
    ]
}

Collection: Address Group
{
    "address_objects": [
        object#address-group-ipv6-config,
        ...
    ]
}

Schema Attributes

address_group:
    Type: object
    Flags: -none-
    Description: Address group.
address_groups:
  Type: array
  Flags: -none-
  Description: Address group collection.

address_group.ipv6:
  Type: object
  Flags: key
  Description: IPV6 address group.

address_group.ipv6.name:
  Type: string
  Flags: key
  Description: Group address object name.

address_group.ipv6.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

address_group.ipv6.address_group:
  Type: object
  Flags: -none-
  Description: Assign address group to group.

address_group.ipv6.address_group.ipv4:
  Type: array
  Flags: -none-
  Description: IPV4 address group.

address_group.ipv6.address_group.ipv4.name:
  Type: string
  Flags: -none-
  Description: Group address object name.

address_group.ipv6.address_group.ipv6:
  Type: array
  Flags: -none-
  Description: IPV6 address group.

address_group.ipv6.address_group.ipv6.name:
  Type: string
  Flags: -none-
  Description: Group address object name.

address_group.ipv6.address_object:
  Type: object
  Flags: -none-
  Description: Assign an IPV6 address object to group.
address_group.ipv6.address_object.ipv4:
  Type: array
  Flags: -none-
  Description: IPV4 address object.

address_group.ipv6.address_object.ipv4.name:
  Type: string
  Flags: -none-
  Description: Host/network/range address object name.

address_group.ipv6.address_object.ipv6:
  Type: array
  Flags: -none-
  Description: IPV6 address object.

address_group.ipv6.address_object.ipv6.name:
  Type: string
  Flags: -none-
  Description: Address object name.

address_group.ipv6.address_object.mac:
  Type: array
  Flags: -none-
  Description: MAC address object.

address_group.ipv6.address_object.mac.name:
  Type: string
  Flags: -none-
  Description: MAC address object name.

address_group.ipv6.address_object.fqdn:
  Type: array
  Flags: -none-
  Description: FQDN address object.

address_group.ipv6.address_object.fqdn.name:
  Type: string
  Flags: -none-
  Description: FQDN address object name.
API: Schedule Objects

- Endpoint on page 42
- Schema Structure on page 42

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/schedules</td>
<td>GET</td>
</tr>
<tr>
<td>HTTP method and body</td>
<td>POST</td>
</tr>
<tr>
<td>Schema: collection#schedule-config</td>
<td>PUT</td>
</tr>
<tr>
<td>Empty</td>
<td>DELETE</td>
</tr>
<tr>
<td>Empty</td>
<td>Required</td>
</tr>
<tr>
<td>Empty</td>
<td>Required</td>
</tr>
<tr>
<td>Empty</td>
<td>Required</td>
</tr>
</tbody>
</table>

## Schema Structure

### Topics:

- **Object: Schedule** on page 42
- **Collection: Schedule** on page 43
- **Schema Attributes** on page 43

### Object: Schedule

```json
{
  "schedule": {
    "name": "{string}",
    "uuid": "{string}",
    "occurs": {
      "once": {
        "event": {
          "start": "{string}",
          "end": "{string}"
        }
      },
      "recurring": [
        "recurring": [
```
"start": "{string}",
"end": "{string}",
"sun": {boolean},
"mon": {boolean},
"tue": {boolean},
"wed": {boolean},
"thu": {boolean},
"fri": {boolean},
"sat": {boolean}
},
...
],
"mixed": {
  "event": {
    "start": "{string}",
    "end": "{string}"  
  },
  "recurring": [
    {
      "start": "{string}",
      "end": "{string}",
      "sun": {boolean},
      "mon": {boolean},
      "tue": {boolean},
      "wed": {boolean},
      "thu": {boolean},
      "fri": {boolean},
      "sat": {boolean}
    },
    ...
  ]
}
}

**Collection: Schedule**

```

"schedules": [
  object#schedule-config,
  ...
]
```

**Schema Attributes**

**schedule:**
- Type: object
- Flags: -none-
- Description: Schedule object.

**schedules:**
- Type: array
- Flags: -none-
Description: Schedule object collection.

**schedule.name:**
Type: string
Flags: key
Description:

**schedule.uuid:**
Type: string
Flags: key
Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

**schedule.occurs:**
Type: object
Flags: -none-
Description: Set schedule type.

**schedule.occurs.once:**
Type: object
Flags: -none-
Description: Set for single occurrence.

**schedule.occurs.once.event:**
Type: object
Flags: -none-
Description: Enter the start and end date and time of a one time event.

**schedule.occurs.once.event.start:**
Type: string (time yyyymmdhhmm)
Flags: -none-
Description: Timestamp in the form: YYYY:MM:DD:HH:MM

**schedule.occurs.once.event.end:**
Type: string (time yyyymmdhhmm)
Flags: -none-
Description: Timestamp in the form: YYYY:MM:DD:HH:MM

**schedule.occurs.recurring:**
Type: object
Flags: -none-
Description: Set for recurring schedule.

**schedule.occurs.recurring.recurring:**
Type: array
Flags: -none-
Description: Add to the list of applicable days and start and stop time of the schedule.

**schedule.occurs.recurring.recurring.start:**
Type: string (time hhmm)
Flags: -none-
Description: Time in the form: DD:DD

schedule.occurs.recurring.recurring.end:
Type: string (time hh:mm)
Flags: -none-
Description: Time in the form: DD:DD

schedule.occurs.recurring.recurring.sun:
Type: boolean (true|false)
Flags: -none-
Description: Day of the week.

schedule.occurs.recurring.recurring.mon:
Type: boolean (true|false)
Flags: -none-
Description: Day of the week.

schedule.occurs.recurring.recurring.tue:
Type: boolean (true|false)
Flags: -none-
Description: Day of the week.

schedule.occurs.recurring.recurring.wed:
Type: boolean (true|false)
Flags: -none-
Description: Day of the week.

schedule.occurs.recurring.recurring.thu:
Type: boolean (true|false)
Flags: -none-
Description: Day of the week.

schedule.occurs.recurring.recurring.fri:
Type: boolean (true|false)
Flags: -none-
Description: Day of the week.

schedule.occurs.recurring.recurring.sat:
Type: boolean (true|false)
Flags: -none-
Description: Day of the week.

schedule.occurs.mixed:
Type: object
Flags: -none-
Description: Set for both recurring schedule and single occurrence.
schedule.occurs.mixed.event:
  Type: object
  Flags: -none-
  Description: Enter the start and end date and time of a one time event.

schedule.occurs.mixed.event.start:
  Type: string (time yyyymmdhhmm)
  Flags: -none-
  Description: Timestamp in the form: YYYY:MM:DD:HH:MM

schedule.occurs.mixed.event.end:
  Type: string (time yyyymmdhhmm)
  Flags: -none-
  Description: Timestamp in the form: YYYY:MM:DD:HH:MM

schedule.occurs.mixed.recurring:
  Type: array
  Flags: -none-
  Description: Add to the list of applicable days and start and stop time of the schedule.

schedule.occurs.mixed.recurring.start:
  Type: string (time hhmm)
  Flags: -none-
  Description: Time in the form: DD:DD

schedule.occurs.mixed.recurring.end:
  Type: string (time hhmm)
  Flags: -none-
  Description: Time in the form: DD:DD

schedule.occurs.mixed.recurring.sun:
  Type: boolean (true|false)
  Flags: -none-
  Description: Day of the week.

schedule.occurs.mixed.recurring.mon:
  Type: boolean (true|false)
  Flags: -none-
  Description: Day of the week.

schedule.occurs.mixed.recurring.tue:
  Type: boolean (true|false)
  Flags: -none-
  Description: Day of the week.

schedule.occurs.mixed.recurring.wed:
  Type: boolean (true|false)
  Flags: -none-
  Description: Day of the week.
schedule.occurs.mixed.recurring.thu:
  Type: boolean (true|false)
  Flags: -none-
  Description: Day of the week.

schedule.occurs.mixed.recurring.fri:
  Type: boolean (true|false)
  Flags: -none-
  Description: Day of the week.

schedule.occurs.mixed.recurring.sat:
  Type: boolean (true|false)
  Flags: -none-
  Description: Day of the week.
API: Service Objects

- **Endpoint** on page 48
- **Schema Structure** on page 48

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/service-objects</td>
<td>GET Empty, POST Required, PUT Required, DELETE Required</td>
</tr>
<tr>
<td>Schema: collection#service-object-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/service-objects/name/{NAME}</td>
<td>GET Empty, PUT Required, DELETE Ignored</td>
</tr>
<tr>
<td>Schema: object#service-object-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/service-objects/uuid/{UUID}</td>
<td>GET Empty, PUT Required, DELETE Ignored</td>
</tr>
<tr>
<td>Schema: object#service-object-config</td>
<td></td>
</tr>
</tbody>
</table>

## Schema Structure

Topics:
- **Object: Service Object** on page 48
- **Collection: Service Object** on page 49
- **Schema Attributes** on page 49

### Object: Service Object

```json
{
    "service_object": {
        "name": "{string}",
        "uuid": "{string}",
        "custom": {number},
        "icmp": "{string}",
        "igmp": "{string}",
        "tcp": {
            "begin": {number},
            "end": {number}
        },
        "udp": {
            "begin": {number},
            "end": {number}
        }
    }
}
```
Collection: Service Object

{   "service-objects": [   object#service-object-config,   ...   ]
}

Schema Attributes

service_object:
Type: object
Flags: -none-
Description: Service object.

service_objects:
Type: array
Flags: -none-
Description: Service object collection.

service_object.name:
Type: string
Flags: key
Description: Service object name.

service_object.uuid:
Type: string
Flags: key
Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH
service_object.custom:
  Type: number (uint8)
  Flags: -none-
  Description: Integer in the form: D OR 0xHH

service_object.icmp:
  Type: string
  Flags: -none-
  Description: Service object ICMP.

service_object.igmp:
  Type: string
  Flags: -none-
  Description: Service object IGMP.

service_object.tcp:
  Type: object
  Flags: -none-
  Description: Service object TCP.

service_object.tcp.begin:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

service_object.tcp.end:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

service_object.udp:
  Type: object
  Flags: -none-
  Description: Service object UDP.

service_object.udp.begin:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

service_object.udp.end:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

service_object.gre:
  Type: boolean (true)
  Flags: -none-
  Description: Service object GRE.
**service_object.esp:**
Type: boolean (true)
Flags: -none-
Description: Service object ESP.

**service_object.6over4:**
Type: boolean (true)
Flags: -none-
Description: Service object 6over4.

**service_object.ah:**
Type: boolean (true)
Flags: -none-
Description: Service object AH.

**service_object.icmpv6:**
Type: string
Flags: -none-
Description: Service object ICMPV6

**service_object.eigrp:**
Type: boolean (true)
Flags: -none-
Description: Service object EIGRP.

**service_object.ospf:**
Type: string
Flags: -none-
Description: Service object OSPF.

**service_object.pim:**
Type: string
Flags: -none-
Description: Service object PIM.

**service_object.l2tp:**
Type: boolean (true)
Flags: -none-
Description: Service object L2TP.

**service_object.ipcomp:**
Type: boolean (true)
Flags: -none-
Description: Service object ipcomp.
API: Service Groups

- **Endpoint** on page 52
- **Schema Structure** on page 52

### Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/service-groups</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#service-group-config</td>
<td>Required</td>
</tr>
<tr>
<td>GET, POST, PUT, DELETE</td>
<td>Required, Required</td>
</tr>
<tr>
<td>URI: /api/sonicos/service-groups/name/{NAME}</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: object#service-group-config</td>
<td>Required, Required</td>
</tr>
<tr>
<td>GET, POST, PUT, DELETE</td>
<td>Required, Required</td>
</tr>
<tr>
<td>URI: /api/sonicos/service-groups/uuid/{UUID}</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: object#service-group-config</td>
<td>Required, Required</td>
</tr>
<tr>
<td>GET, POST, PUT, DELETE</td>
<td>Required, Required</td>
</tr>
</tbody>
</table>

### Schema Structure

**Topics:**
- **Object: Service Group** on page 52
- **Collection: Service Group** on page 53
- **Schema Attributes** on page 53

**Object: Service Group**

```json
{
  "service_group": {
    "name": "{string}"
    "uuid": "{string}"
  },
  "service_object": [{
    "name": "{string}"}
  }, ...
  "service_group": [
  ]
} 
```
Collection: Service Group

```

```

Schema Attributes

service_group:
  Type: object
  Flags: -none-
  Description: Service group.

service_groups:
  Type: array
  Flags: -none-
  Description: Service group collection.

service_group.name:
  Type: string
  Flags: key
  Description: Service object group name.

service_group.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

service_group.service_object:
  Type: array
  Flags: -none-
  Description: Assign service object to group.

service_group.service_object.name:
  Type: string
  Flags: -none-
  Description: Service object name.
service_group.service_group:
   Type: array
   Flags: -none-
   Description: Assign service group to group.

service_group.service_group.name:
   Type: string
   Flags: -none-
   Description: Service object group name.
API: Zones

- **Endpoint** on page 55
- **Schema Structure** on page 55

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/zones</td>
<td>GET Empty POST Required PUT Required DELETE Required</td>
</tr>
<tr>
<td>Schema: collection#zone-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/zones/name/{NAME}</td>
<td>Empty — Required Ignored</td>
</tr>
<tr>
<td>Schema: object#zone-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/zones/uuid/{UUID}</td>
<td>Empty — Required Ignored</td>
</tr>
<tr>
<td>Schema: object#zone-config</td>
<td></td>
</tr>
</tbody>
</table>

## Schema Structure

**Topics:**
- **Object: Zone** on page 55
- **Collection: Zone** on page 57
- **Schema Attributes** on page 57

### Object: Zone

```json
{
    "zone": {
        "name": "{string}",
        "uuid": "{string}",

        "security_type": "{string}",
        "interface_trust": {boolean},

        "auto_generate_access_rules": {
            "allow_from_to_equal": {boolean},
            "allow_from_higher": {boolean},
            "allow_to_lower": {boolean},
            "deny_from_lower": {boolean}
        }
    }
}
```
"websense_content_filtering": {boolean},

"client": {  
    "anti_virus": {boolean},  
    "content_filtering": {boolean}  
},

"gateway_anti_virus": {boolean},  
"intrusion_prevention": {boolean},  
"app_control": {boolean},  
"anti_spyware": {boolean},  
"create_group_vpn": {boolean},  
"ssl_control": {boolean},  
"sslvpn_access": {boolean},

"wireless": {  
    "sslvpn_enforcement": {  
        "server": {  
            "name": "[string]",  
            "host": "[string]"  
        },  
        "service": {  
            "name": "[string]",  
            "protocol": {  
                "name": "[string]",  
                "begin": {number},  
                "end": {number}  
            }  
        }  
    },

    "wifi_sec_enforcement": {  
        "exception_service": {  
            "name": "[string]",  
            "protocol": {  
                "name": "[string]",  
                "begin": {number},  
                "end": {number}  
            }  
        }  
    },

    "wifi_sec_for_site_to_site_vpn": {boolean},  
    "trust_wpa_traffic_as_wifi_sec": {boolean},  
    "only_sonicpoint_traffic": {boolean}  
},

"guest_services": {  
    "inter_guest": {boolean},  

    "bypass": {  
        "client": {  
            "anti_virus": {boolean},  
            "content_filtering": {boolean}  
        }  
    },

    "external_auth": {  
        "client_redirect": "[string]",  
        "web_server": {  
            "protocol": "[string]",  
            "name": "[string]",  
            "port": {number},  
            "ssl": {boolean}  
        }  
    },

    "guest_web_proxy": {  
        "proxy": {boolean},  
        "client": {  
            "anti_virus": {boolean},  
            "content_filtering": {boolean}  
        }  
    }  
}
"timeout": {number}
],

"message_auth": {
    "method": "{string}\",
    "shared_secret": "{string}\",
    "confirm_secret": "{string}\"
},

"social_network": {
    "facebook": {boolean},
    "google": {boolean},
    "twitter": {boolean}
},

"auth_pages": {
    "login": "{string}\",
    "expiration": "{string}\",
    "timeout": "{string}\",
    "max_sessions": "{string}\",
    "traffic_exceeded": "{string}\"
},

"web_content": {
    "redirect": {
        "use_default": {true},
        | "custom": "{string}\"
    },

    "server_down": {
        "use_default": {true},
        | "custom": "{string}\"
    }
}

Collection: Zone
{
    "zones": [
        object#zone-config,
        ...
    ]
}

Schema Attributes

zone:
    Type: object
    Flags: -none-
    Description: Zone object.

zones:
    Type: array
    Flags: -none-
    Description: Zone object collection.
zone.name:
  Type: string
  Flags: key
  Description: Zone object name.

zone.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

zone.security_type:
  Type: string
  Flags: -none-
  Description: Set zone security type.

zone.interface_trust:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable allow interface trust.

zone.auto_generate_access_rules:
  Type: object
  Flags: -none-
  Description: Enable auto generate access rules.

zone.auto_generate_access_rules.allow_from_to_equal:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow traffic between zones with the same trust level.

zone.auto_generate_access_rules.allow_from_higher:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow traffic from zones with higher trust level.

zone.auto_generate_access_rules.allow_to_lower:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow traffic to zones with lower trust level.

zone.auto_generate_access_rules.deny_from_lower:
  Type: boolean (true|false)
  Flags: -none-
  Description: Deny traffic from zones with lower trust level.

zone.websense_content_filtering:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable enforce websense enterprise content filtering service.
zone.client:
  Type: object
  Flags: -none-
  Description: Client settings

zone.client.anti_virus:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable client anti-virus enforcement service.

zone.client.content_filtering:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable client content filtering services enforcement service.

zone.gateway_anti_virus:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable gateway anti-virus service.

zone.intrusion_prevention:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable intrusion prevention service.

zone.app_control:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable app control service.

zone.anti_spyware:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable anti-spyware service.

zone.create_group_vpn:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable automatic creation of group VPN for this zone.

zone.ssl_control:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable SSL-Control on this zone.

zone.sslvpn_access:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable SSL-VPN access this zone.
zone.wireless:
  Type: object
  Flags: -none-
  Description: Enter wireless zone configuration mode.

zone.wireless.sslvpn_enforcement:
  Type: object
  Flags: -none-
  Description: Enable SSLVPN enforcement. Set to null or {} if disabled/unconfigured.

zone.wireless.sslvpn_enforcement.server:
  Type: object
  Flags: -none-
  Description: Set the SSLVPN server as a named address object.

zone.wireless.sslvpn_enforcement.server.name:
  Type: string
  Flags: -none-
  Description: Host address object name.

zone.wireless.sslvpn_enforcement.server.host:
  Type: string (ip)
  Flags: -none-

zone.wireless.sslvpn_enforcement.service:
  Type: object
  Flags: -none-
  Description: Set the SSLVPN service as a named service object.

zone.wireless.sslvpn_enforcement.service.name:
  Type: string
  Flags: -none-
  Description: Service object name.

zone.wireless.sslvpn_enforcement.service.protocol:
  Type: object
  Flags: -none-
  Description: Set the SSLVPN service as a protocol.

zone.wireless.sslvpn_enforcement.service.protocol.name:
  Type: string
  Flags: -none-
  Description: Service protocol.

zone.wireless.sslvpn_enforcement.service.protocol.begin:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH
**zone.wireless.sslvpn_enforcement.service.protocol.end:**
- Type: number (uint16)
- Flags: -none-
- Description: Integer in the form: D OR 0xHHHHH

**zone.wireless.wifi_sec_enforcement:**
- Type: object
- Flags: -none-
- Description: Enable WiFiSec enforcement.

**zone.wireless.wifi_sec_enforcement.exception_service:**
- Type: object
- Flags: -none-
- Description: Specify services that are allowed to bypass wifiSec enforcement.

**zone.wireless.wifi_sec_enforcement.exception_service.name:**
- Type: string
- Flags: -none-
- Description: Service object name.

**zone.wireless.wifi_sec_enforcement.exception_service.protocol:**
- Type: object
- Flags: -none-
- Description: Set the WiFiSec exception service as a protocol.

**zone.wireless.wifi_sec_enforcement.exception_service.protocol.name:**
- Type: string
- Flags: -none-
- Description: Service protocol.

**zone.wireless.wifi_sec_enforcement.exception_service.protocol.begin:**
- Type: number (uint16)
- Flags: -none-
- Description: Integer in the form: D OR 0xHHHHH

**zone.wireless.wifi_sec_enforcement.exception_service.protocol.end:**
- Type: number (uint16)
- Flags: -none-
- Description: Integer in the form: D OR 0xHHHHH

**zone.wireless.wifi_sec_for_site_to_site_vpn:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Enable WiFiSec for site-to-site VPN tunnel traversal.

**zone.wireless.trust_wpa_traffic_as_wifi_sec:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Trust WPA / WPA2 traffic as WiFiSec.
**zone.wireless.only_sonicpoint_traffic:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Enable only allow traffic generated by a SonicPoint/SonicPointN.

**zone.guest_services:**
- Type: object
- Flags: -none-
- Description: Enable zone guest services and enter configuration mode. Set to null or {} if disabled/unconfigured.

**zone.guest_services.inter_guest:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Enable inter-guest communication.

**zone.guest_services.bypass:**
- Type: object
- Flags: -none-
- Description: Enable bypass check for guest clients.

**zone.guest_services.bypass.client:**
- Type: object
- Flags: -none-
- Description: Enable bypass check for guest clients.

**zone.guest_services.bypass.client.anti_virus:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Enable bypass anti-virus check for guests.

**zone.guest_services.bypass.client.content_filtering:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Enable bypass client content filtering check for guests.

**zone.guest_services.external_auth:**
- Type: object
- Flags: -none-
- Description: Enable external guest authentication and enter its configuration mode. Set to null or {} if disabled/unconfigured.

**zone.guest_services.external_auth.client_redirect:**
- Type: string
- Flags: -none-
- Description: Set local web server settings for client redirect.

**zone.guest_services.external_auth.web_server:**
- Type: object
- Flags: -none-
- Description: Configure the external web server settings.
zone.guest_services.external_auth.web_server.protocol:
  Type: string
  Flags: -none-
  Description: Configure the external web server protocol.

zone.guest_services.external_auth.web_server.name:
  Type: string
  Flags: -none-
  Description: FQDN/host address object name.

zone.guest_services.external_auth.web_server.port:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

zone.guest_services.external_auth.web_server.timeout:
  Type: number (uint8)
  Flags: -none-
  Description: Integer in the form: D OR 0xHH

zone.guest_services.external_auth.message_auth:
  Type: object
  Flags: -none-
  Description: Enable external message authentication.

zone.guest_services.external_auth.message_auth.method:
  Type: string
  Flags: -none-
  Description: Set external message authentication method.

zone.guest_services.external_auth.message_auth.shared_secret:
  Type: string
  Flags: -none-
  Description:

zone.guest_services.external_auth.message_auth.confirm_secret:
  Type: string
  Flags: -none-
  Description:

zone.guest_services.external_auth.social_network:
  Type: object
  Flags: -none-
  Description: Enable social network login.

zone.guest_services.external_auth.social_network.facebook:
  Type: boolean (true|false)
  Flags: -none-
Description: Enable Facebook social network login.

`zone.guest_services.external_auth.social_network.google:`
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable Google social network login.

`zone.guest_services.external_auth.social_network.twitter:`
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable Twitter social network login.

`zone.guest_services.external_auth.auth_pages:`
  Type: object
  Flags: -none-
  Description: Configure the external authentication pages.

`zone.guest_services.external_auth.auth_pages.login:`
  Type: string
  Flags: -none-
  Description:

`zone.guest_services.external_auth.auth_pages.expiration:`
  Type: string
  Flags: -none-
  Description:

`zone.guest_services.external_auth.auth_pages.timeout:`
  Type: string
  Flags: -none-
  Description:

`zone.guest_services.external_auth.auth_pages.max_sessions:`
  Type: string
  Flags: -none-
  Description:

`zone.guest_services.external_auth.auth_pages.traffic_exceeded:`
  Type: string
  Flags: -none-
  Description:

`zone.guest_services.external_auth.web_content:`
  Type: object
  Flags: -none-
  Description: Configure the Web content messages.

`zone.guest_services.external_auth.web_content.redirect:`
  Type: object
Flags: -none-
Description: Configure the Web content redirect message.

zone.guest_services.external_auth.web_content.redirect.use_default:
  Type: boolean (true)
  Flags: -none-
  Description: Use the default Web content redirect message.

zone.guest_services.external_auth.web_content.redirect.custom:
  Type: string
  Flags: -none-
  Description:

zone.guest_services.external_auth.web_content.server_down:
  Type: object
  Flags: -none-
  Description: Configure the Web content server down message.

zone.guest_services.external_auth.web_content.server_down.use_default:
  Type: boolean (true)
  Flags: -none-
  Description: Use the default Web content server down message.

zone.guest_services.external_auth.web_content.server_down.custom:
  Type: string
  Flags: -none-
  Description:

zone.guest_services.external_auth.logout_expired:
  Type: object
  Flags: -none-
  Description: Enable auto-session logout.

zone.guest_services.external_auth.logout_expired.every:
  Type: number (uint8)
  Flags: -none-
  Description: Integer in the form: D OR 0xHH

zone.guest_services.external_auth.logout_expired.cgi:
  Type: string
  Flags: -none-
  Description:

zone.guest_services.external_auth.status_check:
  Type: object
  Flags: -none-
  Description: Enable server status check.
### zone.guest_services.external_auth.status_check.every:
- **Type:** number (uint8)
- **Flags:** -none-
- **Description:** Integer in the form: D OR 0xHH

### zone.guest_services.external_auth.status_check.cgi:
- **Type:** string
- **Flags:** -none-
- **Description:**

### zone.guest_services.external_auth.session_sync:
- **Type:** object
- **Flags:** -none-
- **Description:** Enable session synchronization.

### zone.guest_services.external_auth.session_sync.every:
- **Type:** number (uint8)
- **Flags:** -none-
- **Description:** Integer in the form: D OR 0xHH

### zone.guest_services.external_auth.session_sync.cgi:
- **Type:** string
- **Flags:** -none-
- **Description:**

### zone.guest_services.policy_page_non_authentication:
- **Type:** object
- **Flags:** -none-
- **Description:** Enable policy page without authentication and enter its configuration mode. Set to null or {} if disabled/unconfigured.

### zone.guest_services.policy_page_non_authentication.guest_usage_policy:
- **Type:** string
- **Flags:** -none-
- **Description:**

### zone.guest_services.custom_auth_page:
- **Type:** object
- **Flags:** -none-
- **Description:** Enable custom authentication page and enter its configuration mode. Set to null or {} if disabled/unconfigured.

### zone.guest_services.custom_auth_page.header:
- **Type:** object
- **Flags:** -none-
- **Description:** Configure custom page header.

### zone.guest_services.custom_auth_page.header.text:
- **Type:** string
- **Flags:** -none-
- **Description:**
zone.guest_services.custom_auth_page.header.url:
  Type: string (web url)
  Flags: -none-
  Description: URL in the form: http://host/file

zone.guest_services.custom_auth_page.footer:
  Type: object
  Flags: -none-
  Description: Configure custom login page footer.

zone.guest_services.custom_auth_page.footer.text:
  Type: string
  Flags: -none-
  Description:

zone.guest_services.custom_auth_page.footer.url:
  Type: string (web url)
  Flags: -none-
  Description: URL in the form: http://host/file

zone.guest_services.post_auth:
  Type: string (web url)
  Flags: -none-
  Description: URL in the form: http://host/file

zone.guest_services.bypass_guest_auth:
  Type: object
  Flags: -none-
  Description: Enable bypass guest authentication. Set to null or {} if disabled/unconfigured.

zone.guest_services.bypass_guest_auth.all:
  Type: boolean (true)
  Flags: -none-
  Description: All MAC addresses.

zone.guest_services.bypass_guest_auth.name:
  Type: string
  Flags: -none-
  Description: MAC address object name.

zone.guest_services.bypass_guest_auth.group:
  Type: string
  Flags: -none-
  Description: MAC group address object name.

zone.guest_services.bypass_guest_auth.mac:
  Type: string (mac)
  Flags: -none-
zone.guest_services.smtp_redirect:
  Type: object
  Flags: -none-
  Description: Redirect SMTP traffic to specified server. Set to null or {} if disabled/unconfigured.

zone.guest_services.smtp_redirect.name:
  Type: string
  Flags: -none-
  Description: Host address object name.

zone.guest_services.smtp_redirect.host:
  Type: string (ip)
  Flags: -none-

zone.guest_services.deny_networks:
  Type: object
  Flags: -none-
  Description: Enable blocking of traffic to the named network.

zone.guest_services.deny_networks.name:
  Type: string
  Flags: -none-
  Description: Address object name.

zone.guest_services.deny_networks.group:
  Type: string
  Flags: -none-
  Description: Group address object name.

zone.guest_services.deny_networks.mac:
  Type: string (mac)
  Flags: -none-
  Description: Address object MAC address in the form: HH:HH:HH:HH:HH:HH or HHHHHHHHHHHHH or HH-HH-HH-HH-HH-HH.

zone.guest_services.deny_networks.fqdn:
  Type: string (fqdn)
  Flags: -none-
  Description: FQDN in the form: example.com or *.example.com.

zone.guest_services.deny_networks.host:
  Type: string (ip)
  Flags: -none-
zone.guest_services.deny_networks.range:
  Type: object
  Flags: -none-
  Description: Set the denied networks to range of addresses.

zone.guest_services.deny_networks.range.begin:
  Type: string (ip)
  Flags: -none-

zone.guest_services.deny_networks.range.end:
  Type: string (ip)
  Flags: -none-

zone.guest_services.deny_networks.network:
  Type: object
  Flags: -none-
  Description: Set the denied networks to network address.

zone.guest_services.deny_networks.network.subnet:
  Type: string (ip)
  Flags: -none-

zone.guest_services.deny_networks.network.mask:
  Type: string (subnet)
  Flags: -none-
  Description: IPv4 netmask in decimal dotted or CIDR form: D.D.D.D OR /D. IPv6 netmask in the form: /D.

zone.guest_services.deny_networks.ipv6:
  Type: object
  Flags: -none-
  Description: IPv6 address object.

zone.guest_services.deny_networks.ipv6.host:
  Type: string (ip)
  Flags: -none-

zone.guest_services.deny_networks.ipv6.range:
  Type: object
  Flags: -none-
  Description: Set the denied networks to range of addresses.
**zone.guest_services.deny_networks.ipv6.range.begin:**
Type: string (ip)
Flags: -none-

**zone.guest_services.deny_networks.ipv6.range.end:**
Type: string (ip)
Flags: -none-

**zone.guest_services.deny_networks.ipv6.network:**
Type: object
Flags: -none-
Description: Set the denied networks to network address.

**zone.guest_services.deny_networks.ipv6.network.subnet:**
Type: string (ip)
Flags: -none-

**zone.guest_services.deny_networks.ipv6.network.mask:**
Type: string (subnet)
Flags: -none-
Description: IPv4 netmask in decimal dotted or CIDR form: D.D.D.D OR /D. IPv6 netmask in the form: /D.

**zone.guest_services.pass_networks:**
Type: object
Flags: -none-
Description: Enable allowing of traffic to the named network.

**zone.guest_services.pass_networks.name:**
Type: string
Flags: -none-
Description: Address object name.

**zone.guest_services.pass_networks.group:**
Type: string
Flags: -none-
Description: Group address object name.

**zone.guest_services.pass_networks.mac:**
Type: string (mac)
Flags: -none-
Description: Address object MAC address in the form: HHHHHHHHHHHHHH or HHHH-HH-HH-HH-HH-HH.
zone.guest_services.pass_networks.fqdn:
  Type: string (fqdn)
  Flags: -none-
  Description: FQDN in the form: example.com or *.example.com.

zone.guest_services.pass_networks.host:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 host address in the form: D.D.D.D. IPv6 host address in the form:

zone.guest_services.pass_networks.range:
  Type: object
  Flags: -none-
  Description: Set the pass networks to range of addresses.

zone.guest_services.pass_networks.range.begin:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 starting range in the form: D.D.D.D. IPv6 starting range in the form:

zone.guest_services.pass_networks.range.end:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 ending range in the form: D.D.D.D. IPv6 ending range in the form:

zone.guest_services.pass_networks.network:
  Type: object
  Flags: -none-
  Description: Set the pass networks to network address.

zone.guest_services.pass_networks.network.subnet:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 network in the form: D.D.D.D. IPv6 network in the form:

zone.guest_services.pass_networks.network.mask:
  Type: string (subnet)
  Flags: -none-
  Description: IPv4 netmask in decimal dotted or CIDR form: D.D.D.D OR /D. IPv6 netmask in the
  form: /D.

zone.guest_services.pass_networks.ipv6:
  Type: object
  Flags: -none-
  Description: IPv6 address object.
zone.guest_services.pass_networks.ipv6.host:
  Type: string (ip)
  Flags: -none-

zone.guest_services.pass_networks.ipv6.range:
  Type: object
  Flags: -none-
  Description: Set the pass networks to range of addresses.

zone.guest_services.pass_networks.ipv6.range.begin:
  Type: string (ip)
  Flags: -none-

zone.guest_services.pass_networks.ipv6.range.end:
  Type: string (ip)
  Flags: -none-

zone.guest_services.pass_networks.ipv6.network:
  Type: object
  Flags: -none-
  Description: Set the pass networks to network address.

zone.guest_services.pass_networks.ipv6.network.subnet:
  Type: string (ip)
  Flags: -none-

zone.guest_services.pass_networks.ipv6.network.mask:
  Type: string (subnet)
  Flags: -none-
  Description: IPv4 netmask in decimal dotted or CIDR form: D.D.D.D OR /D. IPv6 netmask in the form: /D.

zone.guest_services.max_guests:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

zone.guest_services.dynamic_address_translation:
  Type: boolean (true/false)
  Flags: -none-
  Description: Enable dynamic address translation.
API: DNS

Endpoint

- **Endpoint** on page 73
- **Schema Structure** on page 73

### Endpoint

<table>
<thead>
<tr>
<th>HTTP method and body</th>
<th>GET</th>
<th>POST</th>
<th>PUT</th>
<th>DELETE</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI</td>
<td>/api/sonicos/dns</td>
<td>Empty</td>
<td>—</td>
<td>Required</td>
</tr>
<tr>
<td>Schema</td>
<td>collection#dns-config</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

### Schema Structure

**Topics:**
- **Object: DNS** on page 73
- **Schema Attributes** on page 74

### Object: DNS

```
{
  "dns": {
    "server": {
      "inherit": {true},
      "static": {
        "primary": "[string]",
        "secondary": "[string]",
        "tertiary": "[string]"
      }
    },
    "ipv6": {
      "preferred": {boolean},
      "inherit": {true},
      "static": {
        "primary": "[string]",
        "secondary": "[string]",
        "tertiary": "[string]"
      }
    }
  },
  "rebinding": {
    ...
  }
}
```
Schema Attributes

dns:
  Type: object
  Flags: -none-
  Description: DNS configuration.

dns.server:
  Type: object
  Flags: -none-
  Description: DNS server configuration.

dns.server.inherit:
  Type: boolean (true)
  Flags: -none-
  Description: Inherit DNS servers.

dns.server.static:
  Type: object
  Flags: -none-
  Description: Set static DNS server

dns.server.static.primary:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 host address in the form: D.D.D.D

dns.server.static.secondary:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 host address in the form: D.D.D.D

dns.server.static.tertiary:
  Type: string (ip)
  Flags: -none-
  Description: IPv4 host address in the form: D.D.D.D

dns.server.ipv6:
Flags: -none-
Description: Set IPv6 DNS server

dns.server.ipv6.preferred:
Type: boolean
Flags: -none-
Description: Prefer IPv6 DNS servers.

dns.server.ipv6.inherit:
Type: boolean (true)
Flags: -none-
Description: Inherit DNS servers.

dns.server.ipv6.static:
Type: object
Flags: -none-
Description: Set static DNS server

dns.server.ipv6.static.primary:
Type: string (ip)
Flags: -none-
Description: IPv6 host address in the form: HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH

dns.server.ipv6.static.secondary:
Type: string (ip)
Flags: -none-
Description: IPv6 host address in the form: HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH

dns.server.ipv6.static.tertiary:
Type: string (ip)
Flags: -none-
Description: IPv6 host address in the form: HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH:HHHH

dns.rebinding:
Type: object
Flags: -none-
Description: Enable and configure DNS rebinding attack prevention. Set to null or {} if disabled/unconfigured.

dns.rebinding.action:
Type: string
Flags: -none-
Description: Set action when experiencing attack. Must be one of the following values: log-attack-only | return-query-refused | drop-dns-reply

dns.rebinding.allowed_domains:
Type: object
Flags: -none-
Description: Specify the domains for which checking is not done. Set to null or {} if disabled/unconfigured.
**dns.rebinding.allowed_domains.name:**
- Type: string
- Flags: -none-
- Description: FQDN address object name.

**dns.rebinding.allowed_domains.group:**
- Type: string
- Flags: -none-
- Description: Custom FQDN group address object name.

**dns.fqdn_binding:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Enable FQDN object only cache DNS reply from sanctioned server.
API: Interfaces – IPv4

- Endpoint on page 77
- Schema Structure on page 77

Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/interfaces/ipv4</td>
<td>GET Empty, POST Required, PUT Empty, DELETE —</td>
</tr>
<tr>
<td>Schema: collection#interface-ipv4-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/interfaces/ipv4-config</td>
<td>GET Empty, POST Required, PUT Empty, DELETE —</td>
</tr>
<tr>
<td>Schema: collection#interface-ipv4-config</td>
<td></td>
</tr>
</tbody>
</table>

Schema Structure

Topics:
- Object: Interface – IPv4 on page 77
- Collection: Interface – IPv4 on page 79
- Schema Attributes on page 79

Object: Interface – IPv4

```json
{
  "interface": {
    "ipv4": {
      "name": "{string}",
      "comment": "{string}",
      "ip_assignment": {
        "zone": "{string}",
      },
      "mode": {
        "static": {
          "ip": "{string}",
          "netmask": "{string}",
          "gateway": "{string}",
          "dns": {
            "primary": "{string}",
            "secondary": "{string}",
          }
        },
      }
    }
  }
}
```
"tertiary": "{string}"
},

"backup_ip": "{string}"
},

| "dhcp": {
| "hostname": "{string}"
| "renew_on_startup": {boolean},
| "renew_on_link_up": {boolean},
| "initiate_renewals_with_discover": {boolean},
| "force_discover_interval": {number}
}
}
},

"mtu": {number},

"mac": {
| "default": {true},
| "override": "{string}"
},

"link_speed": {
| "auto_negotiate": {true},
| "half": "{string}"
| "full": "{string}"
},

"management": {
| "http": {boolean},
| "https": {boolean},
| "ping": {boolean},
| "snmp": {boolean},
| "ssh": {boolean}
},

"user_login": {
| "http": {boolean},
| "https": {boolean}
},

"https_redirect": {boolean},
| "send_icmp_fragmentation": {boolean},
| "fragment_packets": {boolean},
| "ignore_df_bit": {boolean},
| "flow_reporting": {boolean},
| "multicast": {boolean},
| "cos_8021p": {boolean},
| "exclude_route": {boolean},
| "asymmetric_route": {boolean},
| "shutdown_port": {boolean},
| "default_8021p_cos": "{string}"
| "policy": "{string}"
},

"sonicpoint": {
| "limit": {number},
| "reserve_address": {
| "dynamic": {true},
| "manual": "{string}"

}}
Collection: Interface – IPv4

```json
{
   "interfaces": [
      object#interface-ipv4-config,
      ...
   ]
}
```

Schema Attributes

**interface:**
Type: object
Flags: -none-
Description: Interface.

**interfaces:**
Type: array
Flags: -none-
Description: Interface collection.

**interface.ipv4:**
Type: object
Flags: -none-
Description: IP version IPV4.

**interface.ipv4.name:**
Type: string
Flags: key
Description: Interface name.

**interface.ipv4.comment:**
Type: string
Flags: -none-
Description:

**interface.ipv4.ip_assignment:**
Type: object
Flags: -none-
Description: Set interface zone and IP assignment. Set to null or {} if disabled/unconfigured.

**interface.ipv4.ip_assignment.zone:**
Type: string
Flags: -none-
Description: Zone object name.

**interface.ipv4.ip_assignment.mode:**
Type: object
Flags: -none-
**Description:** Interface IP assignment mode.

**interface.ipv4.ip_assignment.mode.static:**
- **Type:** object
- **Flags:** -none-
- **Description:** Static IP address assignment.

**interface.ipv4.ip_assignment.mode.static.ip:**
- **Type:** string (v4 ip)
- **Flags:** -none-
- **Description:** IPV4 Address in the form: a.b.c.d

**interface.ipv4.ip_assignment.mode.static.netmask:**
- **Type:** string (v4 subnet)
- **Flags:** -none-
- **Description:** IPV4 netmask in decimal dotted or CIDR form: D.D.D.D OR /D

**interface.ipv4.ip_assignment.mode.static.gateway:**
- **Type:** string (v4 ip)
- **Flags:** -none-
- **Description:** IPV4 Address in the form: a.b.c.d

**interface.ipv4.ip_assignment.mode.static.dns:**
- **Type:** object
- **Flags:** -none-
- **Description:** Set the DNS server IP address.

**interface.ipv4.ip_assignment.mode.static.dns.primary:**
- **Type:** string (v4 ip)
- **Flags:** -none-
- **Description:** IPV4 Address in the form: a.b.c.d

**interface.ipv4.ip_assignment.mode.static.dns.secondary:**
- **Type:** string (v4 ip)
- **Flags:** -none-
- **Description:** IPV4 Address in the form: a.b.c.d

**interface.ipv4.ip_assignment.mode.static.dns.tertiary:**
- **Type:** string (v4 ip)
- **Flags:** -none-
- **Description:** IPV4 Address in the form: a.b.c.d

**interface.ipv4.ip_assignment.mode.static.backup_ip:**
- **Type:** string (v4 ip)
- **Flags:** -none-
- **Description:** IPV4 Address in the form: a.b.c.d

**interface.ipv4.ip_assignment.mode.dhcp:**
- **Type:** object
Flags: -none-
Description: IP address obtained by DHCP.

interface.ipv4.ip_assignment.mode.dhcp.hostname:
  Type: string
  Flags: -none-
  Description:

interface.ipv4.ip_assignment.mode.dhcp.renew_on_startup:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable request renew of previous IP on startup.

interface.ipv4.ip_assignment.mode.dhcp.renew_on_link_up:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable renew DHCP lease on any link up occurrence.

interface.ipv4.ip_assignment.mode.dhcp.initiate_renewals_with_discover:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable initiate renewals with a discover when using DHCP.

interface.ipv4.ip_assignment.mode.dhcp.force_discover_interval:
  Type: number (uint32)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHHHHHHHH

interface.ipv4.mtu:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

interface.ipv4.mac:
  Type: object
  Flags: -none-
  Description: Set MAC address used for this interface.

interface.ipv4.mac.default:
  Type: boolean (true)
  Flags: -none-
  Description: Factory configured MAC.

interface.ipv4.mac.override:
  Type: string (mac)
  Flags: -none-
  Description: MAC address in the form: HH:HH:HH:HH:HH:HH OR HHHHHHHHHHH
interface.ipv4.link_speed:
   Type: object
   Flags: -none-
   Description: Set interface link speed.

interface.ipv4.link_speed.auto_negotiate:
   Type: boolean (true)
   Flags: -none-
   Description: Set interface link speed to auto-negotiate.

interface.ipv4.link_speed.half:
   Type: string
   Flags: -none-
   Description: Half duplex.

interface.ipv4.link_speed.full:
   Type: string
   Flags: -none-
   Description: Full duplex.

interface.ipv4.management:
   Type: object
   Flags: -none-
   Description: Enable management for the specified protocols.

interface.ipv4.management.http:
   Type: boolean (true|false)
   Flags: -none-
   Description: HTTP.

interface.ipv4.management.https:
   Type: boolean (true|false)
   Flags: -none-
   Description: HTTPS.

interface.ipv4.management.ping:
   Type: boolean (true|false)
   Flags: -none-
   Description: Ping.

interface.ipv4.management.snmp:
   Type: boolean (true|false)
   Flags: -none-
   Description: SNMP.

interface.ipv4.management.ssh:
   Type: boolean (true|false)
   Flags: -none-
   Description: SSH.
interface.ipv4.user_login:
  Type: object
  Flags: -none-
  Description: Enable user login for the specified protocols.

interface.ipv4.user_login.http:
  Type: boolean (true|false)
  Flags: -none-
  Description: HTTP.

interface.ipv4.user_login.https:
  Type: boolean (true|false)
  Flags: -none-
  Description: HTTPS.

interface.ipv4.https_redirect:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable redirection from HTTP to HTTPS.

interface.ipv4.send_icmp_fragmentation:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable ICMP fragmentation needed message generation.

interface.ipv4.fragment_packets:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable fragment non-VPN outbound packets larger than this interface's MTU.

interface.ipv4.ignore_df_bit:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable ignore don't fragment (DF) bit.

interface.ipv4.flow_reporting:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable flow reporting on the interface.

interface.ipv4.multicast:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable multicast support.

interface.ipv4.cos_8021p:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable 802.1p support.
interface.ipv4.exclude_route:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable exclude from route advertisement (NSM, OSPF, BGP, RIP).

interface.ipv4.asymmetric_route:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable asymmetric route.

interface.ipv4.shutdown_port:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable shutdown port.

interface.ipv4.default_8021p_cos:
  Type: string
  Flags: -none-
  Description: Enable default 802.1p CoS.

interface.ipv4.policy:
  Type: string
  Flags: -none-
  Description: Tunnel interface VPN policy name.

interface.ipv4.sonicpoint:
  Type: object
  Flags: -none-
  Description: Set SonicPoint parameter.

interface.ipv4.sonicpoint.limit:
  Type: number (uint32)
  Flags: -none-
  Description: SonicPoint limit per interface.

interface.ipv4.sonicpoint.reserve_address:
  Type: object
  Flags: -none-
  Description: Set dynamically or manually reserve SonicPoint address.

interface.ipv4.sonicpoint.reserve_address.dynamic:
  Type: boolean (true)
  Flags: -none-
  Description: Dynamically reserve SonicPoint address.

interface.ipv4.sonicpoint.reserve_address.manual:
  Type: string (v4 ip)
  Flags: -none-
  Description: IPV4 Address in the form: a.b.c.d
API: NAT Policies – IPv4

- **Endpoint** on page 86
- **Schema Structure** on page 86

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/nat-policies/ipv4</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#nat-policies-ipv4-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/nat-policies/ipv4</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#nat-policies-ipv4-config</td>
<td></td>
</tr>
</tbody>
</table>

## Schema Structure

**Topics:**
- **Object: NAT Policies – IPv4** on page 86
- **Collection: NAT Policies – IPv4** on page 88
- **Schema Attributes** on page 88

### Object: NAT Policies – IPv4

```json
{
    "nat_policy": {
        "ipv4": {
            "inbound": "string",
            "outbound": "string",
            "source": {
                "any": {true},
                "name": "string",
                "group": "string"
            },
            "translated_source": {
                "original": {true},
                "name": "string",
                "group": "string"
            }
        }
    }
}
```
"destination": {
  "any": [true],
  | "name": "{string}",
  | "group": "{string}" 
},

"translated_destination": {
  "original": [true],
  | "name": "{string}",
  | "group": "{string}" 
},

"service": {
  "any": [true],
  | "name": "{string}",
  | "group": "{string}" 
},

"translated_service": {
  "original": [true],
  | "name": "{string}",
  | "group": "{string}" 
},

"uuid": "{string}",
"name": "{string}",
"enable": [boolean],
"comment": "{string}",

"priority": {
  "auto": [true],
  | "manual": [number] 
},

"reflexive": [boolean],

"virtual_group": {
  "any": [true],
  | "id": [number] 
},

"nat_method": "{string}",
"source_port_remap": [boolean],

"high_availability": {
  "probing": {
    "probe_every": [number],

    "probe_type": {
      "icmp_ping": [true],
      | "tcp": [number] 
    },

    "reply_timeout": [number],
    "deactivate_after": [number],
    "reactivate_after": [number],
    "rst_as_msis": [boolean],
    "port_probing": [boolean] 
  } 
}
Collection: NAT Policies – IPv4

```json
{
    "nat_policies": [
        object#nat-policy-ipv4-config,
        ...
    ]
}
```

### Schema Attributes

**nat_policy:**
- **Type:** object
- **Flags:** -none-
- **Description:** NAT policy.

**nat_policies:**
- **Type:** array
- **Flags:** -none-
- **Description:** NAT policy collection.

**nat_policy.ipv4:**
- **Type:** object
- **Flags:** -none-
- **Description:** IPv4 NAT policy.

**nat_policy.ipv4.inbound:**
- **Type:** string
- **Flags:** key
- **Description:** Interface name.

**nat_policy.ipv4.outbound:**
- **Type:** string
- **Flags:** key
- **Description:** Interface name.

**nat_policy.ipv4.source:**
- **Type:** object
- **Flags:** key
- **Description:** Specify the original source for the NAT policy.

**nat_policy.ipv4.source.any:**
- **Type:** boolean (true)
- **Flags:** key
- **Description:** Any host.

**nat_policy.ipv4.source.name:**
- **Type:** string
- **Flags:** key
- **Description:** Address object name.
**nat_policy.ipv4.source.group:**
- Type: string
- Flags: key
- Description: Group address object name.

**nat_policy.ipv4.translated_source:**
- Type: object
- Flags: key
- Description: Specify the translated source for the NAT policy.

**nat_policy.ipv4.translated_source.original:**
- Type: boolean (true)
- Flags: key
- Description: Original source IP.

**nat_policy.ipv4.translated_source.name:**
- Type: string
- Flags: key
- Description: Address object name.

**nat_policy.ipv4.translated_source.group:**
- Type: string
- Flags: key
- Description: Group address object name.

**nat_policy.ipv4.destination:**
- Type: object
- Flags: key
- Description: Specify the original destination for the NAT policy.

**nat_policy.ipv4.destination.any:**
- Type: boolean (true)
- Flags: key
- Description: Any host.

**nat_policy.ipv4.destination.name:**
- Type: string
- Flags: key
- Description: Address object name.

**nat_policy.ipv4.destination.group:**
- Type: string
- Flags: key
- Description: Group address object name.

**nat_policy.ipv4.translated_destination:**
- Type: object
- Flags: key
Description: Specify the translated destination for the NAT policy.

**nat_policy.ipv4.translated_destination.original:**
- Type: boolean (true)
- Flags: key
- Description: Original destination IP.

**nat_policy.ipv4.translated_destination.name:**
- Type: string
- Flags: key
- Description: Address object name.

**nat_policy.ipv4.translated_destination.group:**
- Type: string
- Flags: key
- Description: Group address object name.

**nat_policy.ipv4.service:**
- Type: object
- Flags: key
- Description: Specify the original service for the NAT policy.

**nat_policy.ipv4.service.any:**
- Type: boolean (true)
- Flags: key
- Description: Any service.

**nat_policy.ipv4.service.name:**
- Type: string
- Flags: key
- Description: Service object name.

**nat_policy.ipv4.service.group:**
- Type: string
- Flags: key
- Description: Service object group name.

**nat_policy.ipv4.translated_service:**
- Type: object
- Flags: key
- Description: Specify the translated service for the NAT policy.

**nat_policy.ipv4.translated_service.original:**
- Type: boolean (true)
- Flags: key
- Description: Original service.
nat_policy.ipv4.translated_service.name:
  Type: string
  Flags: key
  Description: Service object name.

nat_policy.ipv4.translated_service.group:
  Type: string
  Flags: key
  Description: Service object group name.

nat_policy.ipv4.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

nat_policy.ipv4.name:
  Type: string
  Flags: required
  Description: Name.

nat_policy.ipv4.enable:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable NAT policy.

nat_policy.ipv4.comment:
  Type: string
  Flags: -none-
  Description:

nat_policy.ipv4.priority:
  Type: object
  Flags: -none-
  Description: Set NAT policy priority

  nat_policy.ipv4.priority.auto:
    Type: boolean (true)
    Flags: -none-
    Description: Set auto priority(priority = 0) for NAT policy.

  nat_policy.ipv4.priority.manual:
    Type: number (uint32)
    Flags: -none-
    Description: Integer in the form: D OR 0xHHHHHHHH

nat_policy.ipv4.reflexive:
  Type: boolean (true|false)
  Flags: -none-
  Description: Configure a reflexive rule.
nat_policy.ipv4.virtual_group:
  Type: object
  Flags: -none-
  Description: Specify virtual group for the NAT policy.

nat_policy.ipv4.virtual_group.any:
  Type: boolean (true)
  Flags: -none-
  Description: Any virtual group.

nat_policy.ipv4.virtual_group.id:
  Type: number (uint8)
  Flags: -none-
  Description: Integer in the form: D OR 0xHH

nat_policy.ipv4.nat_method:
  Type: string
  Flags: -none-
  Description: Set the NAT destination translation method.

nat_policy.ipv4.source_port_remap:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable source port remap.

nat_policy.ipv4.high_availability:
  Type: object
  Flags: -none-
  Description: NAT high availability and load balancing configuration mode.

nat_policy.ipv4.high_availability.probing:
  Type: object
  Flags: -none-
  Description: Enable HA probing and enter configuration mode.

nat_policy.ipv4.high_availability.probing.probe_every:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

nat_policy.ipv4.high_availability.probing.probe_type:
  Type: object
  Flags: -none-
  Description: Set probe IP type.

nat_policy.ipv4.high_availability.probing.probe_type.icmp_ping:
  Type: boolean (true)
  Flags: -none-
  Description: ICMP ping probe.
**nat_policy.ipv4.high_availability.probing.probe_type.tcp:**
  - Type: number (uint16)
  - Flags: -none-
  - Description: Integer in the form: D OR 0xHHHHH

**nat_policy.ipv4.high_availability.probing.reply_timeout:**
  - Type: number (uint16)
  - Flags: -none-
  - Description: Integer in the form: D OR 0xHHHHH

**nat_policy.ipv4.high_availability.probing.deactivate_after:**
  - Type: number (uint16)
  - Flags: -none-
  - Description: Integer in the form: D OR 0xHHHHH

**nat_policy.ipv4.high_availability.probing.reactivate_after:**
  - Type: number (uint16)
  - Flags: -none-
  - Description: Integer in the form: D OR 0xHHHHH

**nat_policy.ipv4.high_availability.probing.rst_as_miss:**
  - Type: boolean (true|false)
  - Flags: -none-
  - Description: Enable count RST response as miss.

**nat_policy.ipv4.high_availability.probing.port_probing:**
  - Type: boolean (true|false)
  - Flags: -none-
  - Description: Enable port probing.
API: NAT Policies – IPv6

- Endpoint on page 94
- Schema Structure on page 94

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/nat-policies/ipv6</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#nat-policies-ipv6-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/nat-policies/ipv6</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#nat-policies-ipv6-config</td>
<td></td>
</tr>
</tbody>
</table>

## Schema Structure

Topics:
- Object: NAT Policies – IPv6 on page 94
- Schema Attributes on page 95

### Object: NAT Policies – IPv6

```json
{
   "nat_policy": {
      "ipv6": {
         "inbound": "{string}" ,
         "outbound": "{string}" ,
         "source": {
            "any": {true},
            "name": "{string}" ,
            "group": "{string}" 
         },
         "translated_source": {
            "original": {true},
            "name": "{string}" ,
            "group": "{string}" 
         },
         "destination": {
            "any": {true},
            "name": "{string}" ,
            "group": "{string}" 
         }
      }
   }
}```
| "name": "{string}",  
| "group": "{string}"  
},

"translated_destination": {  
| "original": {true},  
| "name": "{string}",  
| "group": "{string}"  
},

"service": {  
| "any": {true},  
| "name": "{string}",  
| "group": "{string}"  
},

"translated_service": {  
| "original": {true},  
| "name": "{string}",  
| "group": "{string}"  
},

"uuid": "{string}",  
"name": "{string}",  
"enable": {boolean},  
"comment": "{string}",

"priority": {  
| "auto": {true},  
| "manual": {number}  
},

"reflexive": {boolean},

"virtual_group": {  
| "any": {true},  
| "id": {number}  
}
}
}

### Schema Attributes

**nat_policy:**

Type: object  
Flags: -none-  
Description: NAT policy.

**nat_policies:**

Type: object  
Flags: -none-  
Description: NAT policy collection.

**nat_policy.ipv6:**

Type: object  
Flags: key  
Description: IPv6 NAT policy.
nat_policy.ipv6.inbound:
  Type: string
  Flags: key
  Description: Interface name.

nat_policy.ipv6.outbound:
  Type: string
  Flags: key
  Description: Interface name.

nat_policy.ipv6.source:
  Type: object
  Flags: key
  Description: Specify the original source for the NAT policy.

nat_policy.ipv6.source.any:
  Type: boolean (true)
  Flags: key
  Description: Any host.

nat_policy.ipv6.source.name:
  Type: string
  Flags: key
  Description: Address object name.

nat_policy.ipv6.source.group:
  Type: string
  Flags: key
  Description: Group address object name.

nat_policy.ipv6.translated_source:
  Type: object
  Flags: key
  Description: Specify the translated source for the NAT policy.

nat_policy.ipv6.translated_source.original:
  Type: boolean (true)
  Flags: key
  Description: Original source IP.

nat_policy.ipv6.translated_source.name:
  Type: string
  Flags: key
  Description: Address object name.

nat_policy.ipv6.translated_source.group:
  Type: string
  Flags: key
  Description: Group address object name.
nat_policy.ipv6.destination:
Type: object
Flags: key
Description: Specify the original destination for the NAT policy.

nat_policy.ipv6.destination.any:
Type: boolean (true)
Flags: key
Description: Any host.

nat_policy.ipv6.destination.name:
Type: string
Flags: key
Description: Address object name.

nat_policy.ipv6.destination.group:
Type: string
Flags: key
Description: Group address object name.

nat_policy.ipv6.translated_destination:
Type: object
Flags: key
Description: Specify the translated destination for the NAT policy.

nat_policy.ipv6.translated_destination.original:
Type: boolean (true)
Flags: key
Description: Original destination IP.

nat_policy.ipv6.translated_destination.name:
Type: string
Flags: key
Description: Address object name.

nat_policy.ipv6.translated_destination.group:
Type: string
Flags: key
Description: Group address object name.

nat_policy.ipv6.service:
Type: object
Flags: key
Description: Specify the original service for the NAT policy.

nat_policy.ipv6.service.any:
Type: boolean (true)
Flags: key
Description: Any service.
**nat_policy.ipv6.service.name:**
- Type: string
- Flags: key
- Description: Service object name.

**nat_policy.ipv6.service.group:**
- Type: string
- Flags: key
- Description: Service object group name.

**nat_policy.ipv6.translated_service:**
- Type: object
- Flags: key
- Description: Specify the translated service for the NAT policy.

**nat_policy.ipv6.translated_service.original:**
- Type: boolean (true)
- Flags: key
- Description: Original service.

**nat_policy.ipv6.translated_service.name:**
- Type: string
- Flags: key
- Description: Service object name.

**nat_policy.ipv6.translated_service.group:**
- Type: string
- Flags: key
- Description: Service object group name.

**nat_policy.ipv6.uuid:**
- Type: string
- Flags: key
- Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

**nat_policy.ipv6.name:**
- Type: string
- Flags: required
- Description: Name.

**nat_policy.ipv6.enable:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Enable NAT policy.

**nat_policy.ipv6.comment:**
- Type: string
- Flags: -none-
- Description: Policy comment.
**nat_policy.ipv6.priority:**
- Type: object
- Flags: -none-
- Description: Set NAT policy priority.

**nat_policy.ipv6.priority.auto:**
- Type: boolean (true)
- Flags: -none-
- Description: Set auto priority(priority = 0) for NAT policy.

**nat_policy.ipv6.priority.manual:**
- Type: number (uint32)
- Flags: -none-
- Description: Integer in the form: D OR 0xHHHHHHHH

**nat_policy.ipv6.reflexive:**
- Type: boolean (true|false)
- Flags: -none-
- Description: Configure a reflexive rule.

**nat_policy.ipv6.virtual_group:**
- Type: object
- Flags: -none-
- Description: Specify virtual group for the NAT policy.

**nat_policy.ipv6.virtual_group.any:**
- Type: boolean (true)
- Flags: -none-
- Description: Any virtual group.

**nat_policy.ipv6.virtual_group.id:**
- Type: number (uint8)
- Flags: -none-
- Description: Integer in the form: D OR 0xHH
API: NAT Policies – NAT64

**Endpoint**

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/nat-policies/nat64</td>
<td>GET Empty, POST Required, PUT Required, DELETE Required</td>
</tr>
<tr>
<td>Schema: collection#nat-policy-nat64-config</td>
<td></td>
</tr>
</tbody>
</table>

| URI: /api/sonicos/nat-policies/nat64 | GET Empty, PUT Required, DELETE Ignored |
| Schema: collection#nat-policy-nat64-config |

**Schema Structure**

**Topics:**
- **Object:** NAT Policies – NAT64 on page 100
- **Collection:** NAT Policies – NAT64 on page 101
- **Schema Attributes** on page 101

**Object: NAT Policies – NAT64**

```json
{
   "nat_policy": {
      "nat64": {
         "inbound": "{string}",
         "outbound": "{string}",

         "source": {
            "any": {true},
            "name": "{string}",
            "group": "{string}"
         },

         "translated_source": {
            "original": {true},
            "name": "{string}",
            "group": "{string}"
         }
      }
   }
}
```
"pref64": {
    "any": {true},
    "name": "{string}"
},

"translated_destination": {
    "embedded_ipv4_address": {true}
},

"service": {
    "icmp_udp_tcp": {true}
},

"translated_service": {
    "original": {true}
},

"uuid": "{string}",
"name": "{string}"

"enable": {boolean},
"comment": "{string}"

Collection: NAT Policies – NAT64

{ "nat_policies": [ object#nat-policy-nat64-config, ...

] }

Schema Attributes

nat_policy:
Type: object
Flags: -none-
Description: NAT policy.

nat_policies:
Type: object
Flags: -none-
Description: NAT policy collection.

nat_policy.nat64:
Type: object
Flags: key
Description: NAT64 NAT policy.

nat_policy.nat64.inbound:
Type: string
Flags: key
nat_policy.nat64.outbound:
Type: string
Flags: key
Description: Interface name.

nat_policy.nat64.source:
Type: object
Flags: key
Description: Specify the original source for the NAT64 policy.

nat_policy.nat64.source.any:
Type: boolean (true)
Flags: key
Description: Any host.

nat_policy.nat64.source.name:
Type: string
Flags: key
Description: Address object name.

nat_policy.nat64.source.group:
Type: string
Flags: key
Description: Group address object name.

nat_policy.nat64.translated_source:
Type: object
Flags: key
Description: Specify the translated source for the NAT64 policy.

nat_policy.nat64.translated_source.original:
Type: boolean (true)
Flags: key
Description: Original source IP.

nat_policy.nat64.translated_source.name:
Type: string
Flags: key
Description: Address object name.

nat_policy.nat64.translated_source.group:
Type: string
Flags: key
Description: Group address object name.

nat_policy.nat64.pref64:
Flags: key
Description: Specify the prefix for the NAT64 policy.

**nat_policy.nat64.pref64.any:**
Type: boolean (true)
Flags: key
Description: Any host.

**nat_policy.nat64.pref64.name:**
Type: string
Flags: key
Description: Address object name.

**nat_policy.nat64.pref64.group:**
Type: string
Flags: key
Description: Group address object name.

**nat_policy.nat64.translated_destination:**
Type: object
Flags: key
Description: Specify the translated destination for the NAT policy.

**nat_policy.nat64.translated_destination.embedded_ipv4_address:**
Type: boolean (true)
Flags: key
Description: Embedded ipv4 address.

**nat_policy.nat64.service:**
Type: object
Flags: key
Description: Specify the original service for the NAT policy.

**nat_policy.nat64.service.icmp_udp_tcp:**
Type: boolean (true)
Flags: key
Description: ICMP UDP TCP service.

**nat_policy.nat64.translated_service:**
Type: object
Flags: key
Description: Specify the translated service for the NAT policy.

**nat_policy.nat64.translated_service.original:**
Type: boolean (true)
Flags: key
Description: Original service.
nat_policy.nat64.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

nat_policy.nat64.name:
  Type: string
  Flags: required
  Description: Name.

nat_policy.nat64.enable:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable NAT policy.

nat_policy.nat64.comment:
  Type: string
  Flags: -none-
  Description:
API: Access Rules – IPv4

- **Endpoint** on page 105
- **Schema Structure** on page 105

### Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
<th>GET</th>
<th>POST</th>
<th>PUT</th>
<th>DELETE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>URI:</strong> /api/sonicos/access-rules/ipv4</td>
<td>Schema: collection#access-rule-ipv4-config</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>URI:</strong> /api/sonicos/access-rules/ipv4/uuid/{UUID}</td>
<td>Schema: collection#access-rule-ipv4-config</td>
<td>Empty</td>
<td>—</td>
<td>Required</td>
<td>Ignored</td>
</tr>
</tbody>
</table>

### Schema Structure

**Topics:**
- **Object:** Access Rules – IPv4 on page 105
- **Collection:** Access Rules – IPv4 on page 107
- **Schema Attributes** on page 107

**Object: Access Rules – IPv4**

```
{
  "access_rule": {
    "ipv4": {
      "from": "{string}"
    }
  }
```

### Examples

**GET** with an empty body:

```
GET /api/sonicos/access-rules/ipv4 HTTP/1.1
Content-Type: application/json

```

**POST** with required body:

```
POST /api/sonicos/access-rules/ipv4 HTTP/1.1
Content-Type: application/json

```

**PUT** with required body:

```
PUT /api/sonicos/access-rules/ipv4 HTTP/1.1
Content-Type: application/json

```

**DELETE** with required body:

```
DELETE /api/sonicos/access-rules/ipv4 HTTP/1.1
Content-Type: application/json

```
"service": {
  "any": {true},
  "name": "[string]",
  "group": "[string]"
},

"destination": {
  "address": {
    "any": {true},
    "name": "[string]",
    "group": "[string]"
  }
},

"schedule": {
  "always_on": {true},
  "name": "[string]"
},

"users": {
  "included": {
    "all": {true},
    "guests": {true},
    "administrator": {true},
    "name": "[string]",
    "group": "[string]"
  },
  "excluded": {
    "none": {true},
    "guests": {true},
    "administrator": {true},
    "name": "[string]",
    "group": "[string]"
  }
},

"uuid": "[string]",
"name": "[string]",
"comment": "[string]",
"enable": {boolean},
"reflexive": {boolean},
"max_connections": {number},
"logging": {boolean},
"management": {boolean},
"packet_monitoring": {boolean},

"priority": {
  "auto": {true},
  "manual": {number}
},

"tcp": {
  "timeout": {number}
},

"udp": {
  "timeout": {number}
},

"fragments": {boolean},
"botnet_filter": {boolean},
"connection_limit": {
"destination": {
    "threshold": {number}
},

"source": {
    "threshold": {number}
},

"flow_reporting": {boolean},
"geo_ip_filter": {boolean},
"single_sign_on": {boolean},
"cos_override": {boolean},

"quality_of_service": {
    "class_of_service": {
        "explicit": "[string]",
        "map": {true},
        "preserve": {true}
    },
    "dscp": {
        "explicit": {number},
        "map": {true},
        "preserve": {true}
    }
}

Collection: Access Rules – IPv4

{ "access_rules": [
    object#access_rule-ipv4-config,
    ...
] }  

Schema Attributes

access_rule:
  Type: object
  Flags: -none-
  Description: Access rule.

access_rules:
  Type: array
  Flags: -none-
  Description: Access rule collection.

access_rule.ipv4:
  Type: object
  Flags: -none-
  Description: IPv4 access rule.
access_rule.ipv4.from:
  Type: string
  Flags: key
  Description: Zone object name.

access_rule.ipv4.to:
  Type: string
  Flags: key
  Description: Zone object name.

access_rule.ipv4.action:
  Type: string
  Flags: key
  Description: Set the action for this access rule.

access_rule.ipv4.source:
  Type: object
  Flags: key
  Description: Source.

access_rule.ipv4.source.address:
  Type: object
  Flags: key
  Description: Source address.

access_rule.ipv4.source.address.any:
  Type: boolean (true)
  Flags: key
  Description: Any address.

access_rule.ipv4.source.address.name:
  Type: string
  Flags: key
  Description: Address object name.

access_rule.ipv4.source.address.group:
  Type: string
  Flags: key
  Description: Group address object name.

access_rule.ipv4.source.port:
  Type: object
  Flags: key
  Description: Specify a source port for this Access Policy.

access_rule.ipv4.source.port.any:
  Type: boolean (true)
  Flags: key
  Description: Any source service.
access_rule.ipv4.source.port.name:
  Type: string
  Flags: key

Description: Service object name.

access_rule.ipv4.source.port.group:
  Type: string
  Flags: key
  Description: Service object group name.

access_rule.ipv4.service:
  Type: object
  Flags: key
  Description: Specify a destination service for this Access Policy.

access_rule.ipv4.service.any:
  Type: boolean (true)
  Flags: key
  Description: Any destination service.

access_rule.ipv4.service.name:
  Type: string
  Flags: key
  Description: Service object name.

access_rule.ipv4.service.group:
  Type: string
  Flags: key
  Description: Service object group name.

access_rule.ipv4.destination:
  Type: object
  Flags: key
  Description: Destination.

access_rule.ipv4.destination.address:
  Type: object
  Flags: key
  Description: Destination a destination address for this Access Policy.

access_rule.ipv4.destination.address.any:
  Type: boolean (true)
  Flags: key
  Description: Any address.

access_rule.ipv4.destination.address.name:
  Type: string
  Flags: key
  Description: Address object name.
access_rule.ipv4.destination.address.group:
  Type: string
  Flags: key
  Description: Group address object name.

access_rule.ipv4.schedule:
  Type: object
  Flags: key
  Description: Specify a schedule for this access policy.

access_rule.ipv4.schedule.always_on:
  Type: boolean (true)
  Flags: key
  Description: Always on.

access_rule.ipv4.schedule.name:
  Type: string
  Flags: key
  Description: Schedule object name.

access_rule.ipv4.users:
  Type: object
  Flags: key
  Description: Specify users that are excluded from this access policy.

access_rule.ipv4.users.included:
  Type: object
  Flags: key
  Description: Specify included users.

access_rule.ipv4.users.included.all:
  Type: boolean (true)
  Flags: key
  Description: All users.

access_rule.ipv4.users.included.guests:
  Type: boolean (true)
  Flags: key
  Description: Guest users.

access_rule.ipv4.users.included.administrator:
  Type: boolean (true)
  Flags: key
  Description: Administrator.

access_rule.ipv4.users.included.name:
  Type: string
  Flags: key
  Description: Local user object name.
access_rule.ipv4.users.included.group:
  Type: string
  Flags: key
  Description: Local user group object name.

access_rule.ipv4.users.excluded:
  Type: object
  Flags: key
  Description: Specify excluded users.

access_rule.ipv4.users.excluded.none:
  Type: boolean (true)
  Flags: key
  Description: No users.

access_rule.ipv4.users.excluded.guests:
  Type: boolean (true)
  Flags: key
  Description: Guest users.

access_rule.ipv4.users.excluded.administrator:
  Type: boolean (true)
  Flags: key
  Description: Administrator.

access_rule.ipv4.users.excluded.name:
  Type: string
  Flags: key
  Description: Local user object name.

access_rule.ipv4.users.excluded.group:
  Type: string
  Flags: key
  Description: Local user group object name.

access_rule.ipv4.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

access_rule.ipv4.name:
  Type: string
  Flags: required
  Description: Name.

access_rule.ipv4.comment:
  Type: string
  Flags: -none-
  Description:
access_rule.ipv4.enable:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable this access rule.

access_rule.ipv4.reflexive:
  Type: boolean (true|false)
  Flags: -none-
  Description: Configure a reflexive rule.

access_rule.ipv4.max_connections:
  Type: number (uint8)
  Flags: -none-
  Description: Integer in the form: D OR 0xHH

access_rule.ipv4.logging:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable logging when this access rule is used.

access_rule.ipv4.management:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow management traffic.

access_rule.ipv4.packet_monitoring:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable packet monitoring.

access_rule.ipv4.priority:
  Type: object
  Flags: -none-
  Description: Set access rule priority

access_rule.ipv4.priority.auto:
  Type: boolean (true)
  Flags: -none-
  Description: Set auto priority(priority = 0) for access rule.

access_rule.ipv4.priority.manual:
  Type: number (uint32)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHHHHHH

access_rule.ipv4.tcp:
  Type: object
  Flags: -none-
  Description: TCP.
access_rule.ipv4.tcp.timeout:
  Type: number (uint32)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHHHHHH

access_rule.ipv4.udp:
  Type: object
  Flags: -none-
  Description: UDP.

access_rule.ipv4.udp.timeout:
  Type: number (uint32)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHHHHHH

access_rule.ipv4.fragments:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow fragmented packets on this access rule.

access_rule.ipv4.botnet_filter:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable Botnet filter.

access_rule.ipv4.connection_limit:
  Type: object
  Flags: -none-
  Description: Configure connection limit.

access_rule.ipv4.connection_limit.destination:
  Type: object
  Flags: -none-
  Description: Enable connection limit for each destination IP address. Set to null or {} if disabled/unconfigured.

access_rule.ipv4.connection_limit.destination.threshold:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH

access_rule.ipv4.connection_limit.source:
  Type: object
  Flags: -none-
  Description: Enable connection limit for each source IP address. Set to null or {} if disabled/unconfigured.

access_rule.ipv4.connection_limit.source.threshold:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHH
**access_rule.ipv4.flow_reporting:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Enable flow reporting.

**access_rule.ipv4.geo_ip_filter:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Enable Geo-IP filter.

**access_rule.ipv4.single_sign_on:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Invoke single sign on to authenticate users.

**access_rule.ipv4.cos_override:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Allow 802.1p marking to override DSCP values.

**access_rule.ipv4.quality_of_service:**
- **Type:** object
- **Flags:** -none-
- **Description:** Configure quality of service for rule.

**access_rule.ipv4.quality_of_service.class_of_service:**
- **Type:** object
- **Flags:** -none-
- **Description:** Set 802.1p marking action. Set to null or {} if disabled/unconfigured.

**access_rule.ipv4.quality_of_service.class_of_service.explicit:**
- **Type:** string
- **Flags:** -none-
- **Description:** Set explicit marking.

**access_rule.ipv4.quality_of_service.class_of_service.map:**
- **Type:** boolean (true)
- **Flags:** -none-
- **Description:** Map marking.

**access_rule.ipv4.quality_of_service.class_of_service.preserve:**
- **Type:** boolean (true)
- **Flags:** -none-
- **Description:** Preserve marking.

**access_rule.ipv4.quality_of_service.dscp:**
- **Type:** object
- **Flags:** -none-
Description: Set DSCP marking action.

**access_rule.ipv4.quality_of_service.dscp.explicit:**
- **Type:** number (uint8)
- **Flags:** -none-
- **Description:** Integer in the form: D OR 0xHH

**access_rule.ipv4.quality_of_service.dscp.map:**
- **Type:** boolean (true)
- **Flags:** -none-
- **Description:** Map marking.

**access_rule.ipv4.quality_of_service.dscp.preserve:**
- **Type:** boolean (true)
- **Flags:** -none-
- **Description:** Preserve marking.
API: Access Rules – IPv6

Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/access-rules/ipv6</td>
<td>GET Empty, POST Required, PUT Required, DELETE Required</td>
</tr>
<tr>
<td>Schema: collection#access-rule-ipv6-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/access-rules/ipv6/uuid/{UUID}</td>
<td>GET Empty, PUT Required, DELETE Ignored</td>
</tr>
<tr>
<td>Schema: collection#access-rule-ipv6-config</td>
<td></td>
</tr>
</tbody>
</table>

Schema Structure

Topics:
- **Object**: Access Rules – IPv6 on page 116
- **Collection**: Access Rules – IPv6 on page 118
- **Schema Attributes** on page 118

Object: Access Rules – IPv6

```json
{
    "access_rule": {
        "ipv6": {
            "from": "[string]",
            "to": "[string]",
            "action": "[string]",

            "source": {
                "address": {
                    "any": {true},
                    "name": "[string]",
                    "group": "[string]"
                },

                "port": {
                    "any": {true},
                    "name": "[string]",
                    "group": "[string]"
                }
            }
        }
    }
}
```
{  
  "service": {  
    "any": {true},  
    | "name": "[string]",  
    | "group": "[string]"  
  },  
  "destination": {  
    "address": {  
      "any": {true},  
      | "name": "[string]",  
      | "group": "[string]"  
    }  
  },  
  "schedule": {  
    "always_on": {true},  
    | "name": "[string]"  
  },  
  "users": {  
    "included": {  
      | "all": {true},  
      | "guests": {true},  
      | "administrator": {true},  
      | "name": "[string]",  
      | "group": "[string]"  
    },  
    "excluded": {  
      | "none": {true},  
      | "guests": {true},  
      | "administrator": {true},  
      | "name": "[string]",  
      | "group": "[string]"  
    }  
  },  
  "uuid": "[string]",  
  "name": "[string]",  
  "comment": "[string]",  
  "enable": {boolean},  
  "reflexive": {boolean},  
  "max_connections": {number},  
  "logging": {boolean},  
  "management": {boolean},  
  "packet_monitoring": {boolean},  
  "priority": {  
    | "auto": {true},  
    | "manual": {number}  
  },  
  "tcp": {  
    | "timeout": {number}  
  },  
  "udp": {  
    | "timeout": {number}  
  },  
  "fragments": {boolean},  
  "botnet_filter": {boolean},  
  "connection_limit": {  

}
"destination": {
   "threshold": {number}
},

"source": {
   "threshold": {number}
},

"flow_reporting": {boolean},
"geo_ip_filter": {boolean},
"single_sign_on": {boolean},
"cos_override": {boolean},

"quality_of_service": {
   "class_of_service": {
      "explicit": "[string]",
      "map": {true},
      "preserve": {true}
   },

   "dscp": {
      "explicit": {number},
      "map": {true},
      "preserve": {true}
   }
}

Collection: Access Rules – IPv6
{
   "access_rules": [
      object#access_rule-ipv6-config,
      ...
   ]
}

Schema Attributes

access_rule:
   Type: object
   Flags: -none-
   Description: Access rule.

access_rules:
   Type: array
   Flags: -none-
   Description: Access rule collection.

access_rule.ipv6:
   Type: object
   Flags: -none-
   Description: IPv6 access rule.
access_rule.ipv6.from:
  Type: string
  Flags: key
  Description: Zone object name.

access_rule.ipv6.to:
  Type: string
  Flags: key
  Description: Zone object name.

access_rule.ipv6.action:
  Type: string
  Flags: key
  Description: Set the action for this access rule.

access_rule.ipv6.source:
  Type: object
  Flags: key
  Description: Source.

access_rule.ipv6.source.address:
  Type: object
  Flags: key
  Description: Source address.

access_rule.ipv6.source.address.any:
  Type: boolean (true)
  Flags: key
  Description: Any address.

access_rule.ipv6.source.address.name:
  Type: string
  Flags: key
  Description: Address object name.

access_rule.ipv6.source.address.group:
  Type: string
  Flags: key
  Description: Group address object name.

access_rule.ipv6.source.port:
  Type: object
  Flags: key
  Description: Specify a source port for this Access Policy.

access_rule.ipv6.source.port.any:
  Type: boolean (true)
  Flags: key
  Description: Any source service.
access_rule.ipv6.source.port.name:
  Type: string
  Flags: key
  Description: Service object name.

access_rule.ipv6.source.port.group:
  Type: string
  Flags: key
  Description: Service object group name.

access_rule.ipv6.service:
  Type: object
  Flags: key
  Description: Specify a destination service for this Access Policy.

access_rule.ipv6.service.any:
  Type: boolean (true)
  Flags: key
  Description: Any destination service.

access_rule.ipv6.service.name:
  Type: string
  Flags: key
  Description: Service object name.

access_rule.ipv6.service.group:
  Type: string
  Flags: key
  Description: Service object group name.

access_rule.ipv6.destination:
  Type: object
  Flags: key
  Description: Destination.

access_rule.ipv6.destination.address:
  Type: object
  Flags: key
  Description: Destination a destination address for this Access Policy.

access_rule.ipv6.destination.address.any:
  Type: boolean (true)
  Flags: key
  Description: Any address.

access_rule.ipv6.destination.address.name:
  Type: string
  Flags: key
  Description: Address object name.
access_rule.ipv6.destination.address.group:
  Type: string
  Flags: key
  Description: Group address object name.

access_rule.ipv6.schedule:
  Type: object
  Flags: key
  Description: Specify a schedule for this access policy.

access_rule.ipv6.schedule.always_on:
  Type: boolean (true)
  Flags: key
  Description: Always on.

access_rule.ipv6.schedule.name:
  Type: string
  Flags: key
  Description: Schedule object name.

access_rule.ipv6.users:
  Type: object
  Flags: key
  Description: Specify users that are excluded from this access policy.

access_rule.ipv6.users.included:
  Type: object
  Flags: key
  Description: Specify included users.

access_rule.ipv6.users.included.all:
  Type: boolean (true)
  Flags: key
  Description: All users.

access_rule.ipv6.users.included.guests:
  Type: boolean (true)
  Flags: key
  Description: Guest users.

access_rule.ipv6.users.included.administrator:
  Type: boolean (true)
  Flags: key
  Description: Administrator.

access_rule.ipv6.users.included.name:
  Type: string
  Flags: key
  Description: Local user object name.
access_rule.ipv6.users.included.group:
  Type: string
  Flags: key
  Description: Local user group object name.

access_rule.ipv6.users.excluded:
  Type: object
  Flags: key
  Description: Specify excluded users.

access_rule.ipv6.users.excluded.none:
  Type: boolean (true)
  Flags: key
  Description: No users.

access_rule.ipv6.users.excluded.guests:
  Type: boolean (true)
  Flags: key
  Description: Guest users.

access_rule.ipv6.users.excludedadministrator:
  Type: boolean (true)
  Flags: key
  Description: Administrator.

access_rule.ipv6.users.excluded.name:
  Type: string
  Flags: key
  Description: Local user object name.

access_rule.ipv6.users.excluded.group:
  Type: string
  Flags: key
  Description: Local user group object name.

access_rule.ipv6.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

access_rule.ipv6.name:
  Type: string
  Flags: required
  Description: Name.

access_rule.ipv6.comment:
  Type: string
  Flags: -none-
  Description:
access_rule.ipv6.enable:
   Type: boolean (true|false)
   Flags: -none-
   Description: Enable this access rule.

access_rule.ipv6.reflexive:
   Type: boolean (true|false)
   Flags: -none-
   Description: Configure a reflexive rule.

access_rule.ipv6.max_connections:
   Type: number (uint8)
   Flags: -none-
   Description: Integer in the form: D OR 0xHH

access_rule.ipv6.logging:
   Type: boolean (true|false)
   Flags: -none-
   Description: Enable logging when this access rule is used.

access_rule.ipv6.management:
   Type: boolean (true|false)
   Flags: -none-
   Description: Allow management traffic.

access_rule.ipv6.packet_monitoring:
   Type: boolean (true|false)
   Flags: -none-
   Description: Enable packet monitoring.

access_rule.ipv6.priority:
   Type: object
   Flags: -none-
   Description: Set access rule priority

access_rule.ipv6.priority.auto:
   Type: boolean (true)
   Flags: -none-
   Description: Set auto priority(priority = 0) for access rule.

access_rule.ipv6.priority.manual:
   Type: number (uint32)
   Flags: -none-
   Description: Integer in the form: D OR 0xHHHHHHHH

access_rule.ipv6.tcp:
   Type: object
   Flags: -none-
   Description: TCP.
access_rule.ipv6.tcp.timeout:
  Type: number (uint32)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHHHHHH

access_rule.ipv6.udp:
  Type: object
  Flags: -none-
  Description: UDP.

access_rule.ipv6.udp.timeout:
  Type: number (uint32)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHHHHHHH

access_rule.ipv6.fragments:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow fragmented packets on this access rule.

access_rule.ipv6.botnet_filter:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable Botnet filter.

access_rule.ipv6.connection_limit:
  Type: object
  Flags: -none-
  Description: Configure connection limit.

access_rule.ipv6.connection_limit.destination:
  Type: object
  Flags: -none-
  Description: Enable connection limit for each destination IP address. Set to null or {} if disabled/unconfigured.

access_rule.ipv6.connection_limit.destination.threshold:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHH

access_rule.ipv6.connection_limit.source:
  Type: object
  Flags: -none-
  Description: Enable connection limit for each source IP address. Set to null or {} if disabled/unconfigured.

access_rule.ipv6.connection_limit.source.threshold:
  Type: number (uint16)
  Flags: -none-
  Description: Integer in the form: D OR 0xHHH
access_rule.ipv6.flow_reporting:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable flow reporting.

access_rule.ipv6.geo_ip_filter:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable Geo-IP filter.

access_rule.ipv6.single_sign_on:
  Type: boolean (true|false)
  Flags: -none-
  Description: Invoke single sign on to authenticate users.

access_rule.ipv6.cos_override:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow 802.1p marking to override DSCP values.

access_rule.ipv6.quality_of_service:
  Type: object
  Flags: -none-
  Description: Configure quality of service for rule.

access_rule.ipv6.quality_of_service.class_of_service:
  Type: object
  Flags: -none-
  Description: Set 802.1p marking action. Set to null or {} if disabled/unconfigured.

access_rule.ipv6.quality_of_service.class_of_service.explicit:
  Type: string
  Flags: -none-
  Description: Set explicit marking.

access_rule.ipv6.quality_of_service.class_of_service.map:
  Type: boolean (true)
  Flags: -none-
  Description: Map marking.

access_rule.ipv6.quality_of_service.class_of_service.preserve:
  Type: boolean (true)
  Flags: -none-
  Description: Preserve marking.

access_rule.ipv6.quality_of_service.dscp:
  Type: object
  Flags: -none-
Description: Set DSCP marking action.

**access_rule.ipv6.quality_of_service.dscp.explicit:**
- Type: number (uint8)
- Flags: -none-
- Description: Integer in the form: D OR 0xHH

**access_rule.ipv6.quality_of_service.dscp.map:**
- Type: boolean (true)
- Flags: -none-
- Description: Map marking.

**access_rule.ipv6.quality_of_service.dscp.preserve:**
- Type: boolean (true)
- Flags: -none-
- Description: Preserve marking.
API: Route Policies – IPv4

- **Endpoint** on page 127
- **Schema Structure** on page 127

**Endpoint**

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/route-policies/ipv4</td>
<td>GET Empty, POST Required, PUT Required, DELETE Required</td>
</tr>
<tr>
<td>Schema: collection#route-policy-ipv4-config</td>
<td></td>
</tr>
<tr>
<td>URI: /api/sonicos/route-policies/ipv4/uuid/{UUID}</td>
<td>GET Empty, PUT Required, DELETE Ignored</td>
</tr>
<tr>
<td>Schema: collection#route-policy-ipv4-config</td>
<td></td>
</tr>
</tbody>
</table>

**Schema Structure**

**Topics:**

- **Object: Route Policies – IPv4** on page 127
- **Collection: Route Policies – IPv4** on page 128
- **Schema Attributes** on page 128

**Object: Route Policies – IPv4**

```json
{
    "route_policy": {
        "ipv4": {
            "interface": "{string}",
            "metric": {number},

            "source": {
                "any": {true},
                | "name": "{string}",
                | "group": "{string}
            },

            "destination": {
                "any": {true},
                | "name": "{string}",
                | "group": "{string}
            }
        }
    }
}
```

SonicWall SonicOS API 6.5.1 Reference
API: Route Policies – IPv4 | 127
"service": {
  "any": {true},
  | "name": "{string}"
  | "group": "{string}"
},

"gateway": {
  "default": {true},
  | "name": "{string}"
  | "host": "{string}"
},

"uuid": "{string}"
"name": "{string}"
"disable_on_interface_down": {boolean}
"vpn_precedence": {boolean}
"auto_add_access_rules": {boolean}
"probe": "{string}"
"disable_when_probes_succeed": {boolean}
"default_probe_state_up": {boolean}
"comment": "{string}"
"tcp_acceleration": {boolean}
"wxa_group": "{string}"
}

Collection: Route Policies – IPv4
{
  "route_policies": [
    object#route-policy-ipv4-config,
    ...
  ]
}

Schema Attributes

route_policy:
  Type: object
  Flags: -none-
  Description: Route policy.

route_policies:
  Type: array
  Flags: -none-
  Description: Route policy collection.

route_policy.ipv4:
  Type: object
  Flags: -none-
  Description: IPv4 route policy.
route_policy.ipv4.interface:
  Type: string
  Flags: key
  Description: Route interface name.

route_policy.ipv4.metric:
  Type: number (uint8)
  Flags: key
  Description: Integer in the form: D OR 0xHH

route_policy.ipv4.source:
  Type: object
  Flags: key
  Description: Set route policy source.

route_policy.ipv4.source.any:
  Type: boolean (true)
  Flags: key
  Description: Any host.

route_policy.ipv4.source.name:
  Type: string
  Flags: key
  Description: Host/network/range address object name.

route_policy.ipv4.source.group:
  Type: string
  Flags: key
  Description: Group address object name.

route_policy.ipv4.destination:
  Type: object
  Flags: key
  Description: Set route policy destination.

route_policy.ipv4.destination.any:
  Type: boolean (true)
  Flags: key
  Description: Any host.

route_policy.ipv4.destination.name:
  Type: string
  Flags: key
  Description: FQDN/host/network/range address object name.

route_policy.ipv4.destination.group:
  Type: string
  Flags: key
  Description: Group address object name.
route_policy.ipv4.service:
    Type: object
    Flags: key
    Description: Set route policy service.

route_policy.ipv4.service.any:
    Type: boolean (true)
    Flags: key
    Description: Any service.

route_policy.ipv4.service.name:
    Type: string
    Flags: key
    Description: Service object name.

route_policy.ipv4.service.group:
    Type: string
    Flags: key
    Description: Service object group name.

route_policy.ipv4.gateway:
    Type: object
    Flags: key
    Description: Set route policy gateway.

route_policy.ipv4.gateway.default:
    Type: boolean (true)
    Flags: key
    Description: Default gateway 0.0.0.0

route_policy.ipv4.gateway.name:
    Type: string
    Flags: key
    Description: Host address object name.

route_policy.ipv4.gateway.host:
    Type: string (ip)
    Flags: key
    Description: IPv4 host address in the form: D.D.D.D

route_policy.ipv4.uuid:
    Type: string
    Flags: key
    Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

route_policy.ipv4.name:
    Type: string
    Flags: required
    Description: Name.
route_policy.ipv4.disable_on_interface_down:
  Type: boolean (true|false)
  Flags: -none-
  Description: Disable route when the interface is disconnected.

route_policy.ipv4.vpn_precedence:
  Type: boolean (true|false)
  Flags: -none-
  Description: Allow VPN path to take precedence.

route_policy.ipv4.auto_add_access_rules:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable auto-add access rules.

route_policy.ipv4.probe:
  Type: string
  Flags: -none-
  Description: Atom Object name.

route_policy.ipv4.disable_when_probes_succeed:
  Type: boolean (true|false)
  Flags: -none-
  Description: Disable route when probe succeeds.

route_policy.ipv4.default_probe_state_up:
  Type: boolean (true|false)
  Flags: -none-
  Description: Set probe default state to up.

route_policy.ipv4.comment:
  Type: string
  Flags: -none-
  Description:

route_policy.ipv4.tcp_acceleration:
  Type: boolean (true|false)
  Flags: -none-
  Description: Enable permit TCP acceleration.

route_policy.ipv4.wxa_group:
  Type: string
  Flags: -none-
  Description: WXA group name.
API: Route Policies – IPv6

- **Endpoint** on page 132
- **Schema Structure** on page 132

## Endpoint

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>HTTP method and body</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI: /api/sonicos/route-policies/ipv6</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#route-policy-ipv6-config</td>
<td>Required</td>
</tr>
<tr>
<td>URI: /api/sonicos/route-policies/ipv6/uuid/{UUID}</td>
<td>Empty</td>
</tr>
<tr>
<td>Schema: collection#route-policy-ipv6-config</td>
<td></td>
</tr>
</tbody>
</table>

## Schema Structure

### Topics:
- **Object: Route Policies – IPv6** on page 132
- **Collection: Route Policies – IPv6** on page 133
- **Schema Attributes** on page 133

### Object: Route Policies – IPv6

```json
{
    "route_policy": {
        "ipv6": {
            "interface": "{string}",
            "metric": {number},

            "source": {
                "any": {true},
                | "name": "{string}",
                | "group": "{string}"
            },

            "destination": {
                "any": {true},
                | "name": "{string}",
                | "group": "{string}"
            }
        }
    }
}
```
"service": {
    "any": {true},
    | "name": "{string}",
    | "group": "{string}" 
},

"gateway": {
    "default": {true},
    | "name": "{string}",
    | "host": "{string}" 
},

"uuid": "{string}",
"name": "{string}",
"disable_on_interface_down": {boolean},
"vpn_precedence": {boolean},
"auto_add_access_rules": {boolean},
"probe": "{string}",
"disable_when_probes_succeed": {boolean},
"default_probe_state_up": {boolean},
"comment": "{string}" 

Collection: Route Policies – IPv6

{ "route_policies": [ 
    object#route-policy-ipv6-config,
    ... 
] }

Schema Attributes

route_policy:
  Type: object
  Flags: -none-
  Description: Route policy.

route_policies:
  Type: array
  Flags: -none-
  Description: Route policy collection.

route_policy.ipv6:
  Type: object
  Flags: key
  Description: IPv6 route policy.

route_policy.ipv6.interface:
  Type: string
  Flags: key
  Description: Route interface name.
route_policy.ipv6.metric:
  Type: number (uint8)
  Flags: key
  Description: Integer in the form: D OR 0xHH

route_policy.ipv6.source:
  Type: object
  Flags: key
  Description: Set route policy source.

route_policy.ipv6.source.any:
  Type: boolean (true)
  Flags: key
  Description: Any host.

route_policy.ipv6.source.name:
  Type: string
  Flags: key
  Description: Host/network/range address object name.

route_policy.ipv6.source.group:
  Type: string
  Flags: key
  Description: Group address object name.

route_policy.ipv6.destination:
  Type: object
  Flags: key
  Description: Set route policy destination.

route_policy.ipv6.destination.any:
  Type: boolean (true)
  Flags: key
  Description: Any host.

route_policy.ipv6.destination.name:
  Type: string
  Flags: key
  Description: FQDN/host/network/range address object name.

route_policy.ipv6.destination.group:
  Type: string
  Flags: key
  Description: Group address object name.
Description: Set route policy service.

route_policy.ipv6.service.any:
  Type: boolean (true)
  Flags: key
  Description: Any service.

route_policy.ipv6.service.name:
  Type: string
  Flags: key
  Description: Service object name.

route_policy.ipv6.service.group:
  Type: string
  Flags: key
  Description: Service object group name.

route_policy.ipv6.gateway:
  Type: object
  Flags: key
  Description: Set route policy gateway.

route_policy.ipv6.gateway.default:
  Type: boolean (true)
  Flags: key
  Description: Default gateway 0.0.0/::

route_policy.ipv6.gateway.name:
  Type: string
  Flags: key
  Description: Host address object name.

route_policy.ipv6.gateway.host:
  Type: string (ip)
  Flags: key

route_policy.ipv6.uuid:
  Type: string
  Flags: key
  Description: UUID in the form: HHHHHHHH-HHHH-HHHH-HHHH-HHHHHHHHHHHH

route_policy.ipv6.name:
  Type: string
  Flags: required
  Description: Name.
**route_policy.ipv6.disable_on_interface_down:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Disable route when the interface is disconnected.

**route_policy.ipv6.vpn_precedence:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Allow VPN path to take precedence.

**route_policy.ipv6.auto_add_access_rules:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Enable auto-add access rules.

**route_policy.ipv6.probe:**
- **Type:** string
- **Flags:** -none-
- **Description:** Atom Object name.

**route_policy.ipv6.disable_when_probes_succeed:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Disable route when probe succeeds.

**route_policy.ipv6.default_probe_state_up:**
- **Type:** boolean (true|false)
- **Flags:** -none-
- **Description:** Set probe default state to up.

**route_policy.ipv6.comment:**
- **Type:** string
- **Flags:** -none-
- **Description:**
Technical support is available to customers who have purchased SonicWall products with a valid maintenance contract and to customers who have trial versions.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. To access the Support Portal, go to https://www.sonicwall.com/support.

The Support Portal enables you to:

- View knowledge base articles and technical documentation
- View video tutorials
- Access MySonicWall
- Learn about SonicWall professional services
- Review SonicWall Support services and warranty information
- Register for training and certification
- Request technical support or customer service

To contact SonicWall Support, visit https://www.sonicwall.com/support/contact-support.
About This Document

Legend

⚠️ WARNING: A WARNING icon indicates a potential for property damage, personal injury, or death.

⚠️ CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

까요 IMPORTANT, NOTE, TIP, MOBILE, or VIDEO: An information icon indicates supporting information.

Sonicsos Reference
Updated - February 2018
Software Version - 6.5.1
232-001810-00 Rev A

Copyright © 2018 SonicWall Inc. All rights reserved.

SonicWall is a trademark or registered trademark of SonicWall Inc. and/or its affiliates in the U.S.A. and/or other countries. All other trademarks and registered trademarks are property of their respective owners.

The information in this document is provided in connection with SonicWall Inc. and/or its affiliates' products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of SonicWall products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, SONICWALL AND/OR ITS AFFILIATES ASSUME NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL SONICWALL AND/OR ITS AFFILIATES BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF SONICWALL AND/OR ITS AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SonicWall and/or its affiliates make no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. SonicWall Inc. and/or its affiliates do not make any commitment to update the information contained in this document.

For more information, visit https://www.sonicwall.com/legal.

End User Product Agreement
To view the SonicWall End User Product Agreement, go to: https://www.sonicwall.com/en-us/legal/license-agreements. Select the language based on your geographic location to see the EUPA that applies to your region.

Open Source Code
SonicWall is able to provide a machine-readable copy of open source code with restrictive licenses such as GPL, LGPL, AGPL when applicable per license requirements. To obtain a complete machine-readable copy, send your written requests, along with certified check or money order in the amount of USD 25.00 payable to "SonicWall Inc.", to:

General Public License Source Code Request
SonicWall Inc. Attn: Jennifer Anderson
1033 McCarthy Blvd
Milpitas, CA 95035