

TR-064 Support – AppSetup

Author: FRITZ! GmbH

Date: 2025-08-07

1 urn:X_AVM-DE_AppSetup-com:serviceId:X_AVM-DE_AppSetup1

For details please refer the SCPD files on the device.

Content

| | | |
|------|--|----|
| 1 | urn:X_AVM-DE_AppSetup-com:serviceId:X_AVM-DE_AppSetup1 | 1 |
| 2 | History | 2 |
| 3 | Action List | 3 |
| 3.1 | GetAppMessageFilter | 3 |
| 3.2 | GetAppRemoteInfo | 4 |
| 3.3 | GetConfig | 4 |
| 3.4 | GetInfo | 5 |
| 3.5 | RegisterApp | 6 |
| 3.6 | ResetEvent | 8 |
| 3.7 | SetAppMessageFilter | 9 |
| 3.8 | SetAppMessageReceiver | 10 |
| 3.9 | SetAppVPN | 11 |
| 3.10 | SetAppVPNwithPFS | 12 |
| 3.11 | GetBoxSenderId | 13 |
| 4 | Parameters | 14 |
| 4.1 | RightEnum | 14 |
| 4.2 | EventId | 14 |
| 5 | Service States Table | 15 |

2 History

| Date | Version | Changes |
|------------|---------|---|
| 2015-02-23 | 8 | Parameter Port and DynDNSName only in action GetDynDNS |
| 2015-03-13 | 9 | Remove out parameter ReturnCode from Action RegisterApp Add ReturnCodes for Action RegisterApp Remove Action GetDynDns Remove Action SetAppVoIP |
| 2015-04-02 | 10 | Add ReturnCodes for Action RegisterApp Add Service State Table |
| 2015-06-11 | 11 | Action SetAppMessageReceiver added |
| 2015-06-22 | 12 | Action SetAppMessageReceiver changed |
| 2015-07-06 | 13 | Move chapter Fehler: Verweis nicht gefunden Fehler: Verweis nicht gefundento 3 Action List Add parameters of SetAppMessageReceiver to chapter 5 Service States Table Change format |
| 2015-07-17 | 14 | Add ReturnCodes for action SetAppVPN |
| 2015-07-17 | 15 | SetAppMessageReceiver: Updated description, parameter and return codes |
| 2015-10-01 | 16 | Add description to delete VPN configuration with action SetAppVPN in chapter 3.9 SetAppVPN . |
| 2016-01-04 | 17 | SetAppMessageReceiver – AppAVMPasswordHash has to be base64url encoded without padding |
| 2016-10-27 | 18 | Add action ResetEvent and parameter EventId |
| 2017-11-09 | 19 | RegisterApp() → add new ReturnCodes (841) RegisterApp() → corrected rights in documentation |
| 2018-03-21 | 20 | New action SetAppVPNwithPFS added. New action SetAppMessageFilter and GetAppMessageFilter added. Sort actions in alphabetic order. Add return code for action GetConfig. |
| 2018-04-10 | 21 | Add parameters MinCharsFilter, MaxCharsFilter and AllowedCharsFilter. |
| 2019-02-08 | 22 | New action GetAppRemoteInfo() New Statevariables: SubnetMask, IPAddress, ExternalIPAddress, ExternalPv6Address, RemoteAccessDDNSEnabled, RemoteAccessDDNSDomain, MyFritzDynDNSEnabled, MyFritzDynDNSName |
| 2019-04-08 | 23 | Added return codes to action RegisterApp, GetAppMessageFilter |
| 2020-09-03 | 24 | Added Legacy vs Changed behavior description for action GetAppRemoteInfo() |
| 2022-02-02 | 25 | Add GetBoxSenderId() |
| 2022-10-17 | 26 | Added allowed values for ConfigRight, AppRight, NasRight, PhoneRight, DialRight, HomeautoRight |
| 2025-08-07 | 27 | Changed AVM to FRITZ! |

3 Action List

This chapter contains the supported actions of the service AppSetup which are listed incl. arguments.

The name of the argument name always uses the 'New' prefix.

3.1 GetAppMessageFilter

Returns a list of all message-filters configured for an app instance.

- Required rights: any
- The security context of this action **must belong to an app instance**. The app instance of the security context must be the same as the one identified by the parameter AppId.

| Argument name | Direction | Meaning | Data type |
|---------------|-----------|--------------------------------------|-----------|
| AppId | In | Identifier of the app instance | string |
| FilterList | out | The result of action in a XML format | string |

Table 1: Argument list of action GetAppMessageFilter

| Return code | Description | Related argument |
|-------------|--------------------------------------|------------------|
| 402 | Invalid arguments | Any |
| 501 | Action failed | |
| 603 | Device is out of memory | Any |
| 606 | Unauthorized. Wrong security context | AppId |
| 714 | App with AppId not found | AppId |
| 801 | String too short | AppId |
| 802 | String too long | AppId |
| 803 | Argument contains invalid characters | AppId |
| 820 | Internal error | |

Table 2: Return codes of action GetAppMessageFilter

3.2 GetAppRemoteInfo

Return information which is needed for apps to have access to the FRITZ!Box via WAN.

Legacy behavior:

Either ExternalIPAddress or ExternalIPv6Address has returned a valid address. If WAN interface has both addresses, only IPv4 was returned.

Changed behavior:

If WAN interface has IPv4 and IPv6 addresses, both parameter return a valid address.

E.g. to create and handle a vpn connection.

- Required rights: Config or App

| Argument name | Direction | Meaning | Data type |
|-------------------------|-----------|---|-----------|
| SubnetMask | out | Subnetmask of the local CPE network | string |
| IPAddress | out | Local IP address of the CPE device | string |
| ExternalIPAddress | out | IP address of the WAN interface | string |
| ExternalIPv6Address | out | IPv6 address of the WAN interface | string |
| RemoteAccessDDNSEnabled | out | Shows if a specific DynDNS is activated | boolean |
| RemoteAccessDDNSDomain | out | Domain of the DynDNS | string |
| MyFritzDynDNSEnabled | out | Shows if the MyFritz of AVM GmbH is activated | boolean |
| MyFritzDynDNSName | out | MyFritz URL | string |

Table 3: Argument list of action GetAppRemoteInfo

| Return code | Description | Related argument |
|-------------|-------------------|------------------|
| 402 | Invalid arguments | Any |
| 820 | Internal error | |

Table 4: Return codes of action GetAppRemoteInfo

3.3 GetConfig

This action queries the current access rights of the current TR-064 security context. This security context results from the authentication of a box user or an app instance.

- Required rights: any

| Argument name | Direction | Meaning | Data type |
|--------------------|-----------|---|-----------|
| ConfigRight | out | FRITZ!OS configuration right. | RightEnum |
| AppRight | out | FRITZ!OS app specific configuration right. | RightEnum |
| NasRight | out | FRITZ!OS NAS specific right. | RightEnum |
| PhoneRight | out | FRITZ!OS phone specific right. | RightEnum |
| DialRight | out | FRITZ!OS dial specific right. | RightEnum |
| HomeautoRight | out | FRITZ!OS home automation specific right. | RightEnum |
| InternetRights | out | Are access rights from the internet configured? | boolean |
| AccessFromInternet | out | Is the current access coming from the internet? | boolean |

Table 5: Argument list of action GetConfig

| Return code | Description | Related argument |
|-------------|---------------|------------------|
| 501 | Action failed | |

Table 6: Return codes of action GetConfig

3.4 GetInfo

Read restrictive values for action parameters.

- Required rights: any

| Argument name | Direction | Meaning | Data type |
|--------------------------------|-----------|---|-----------|
| MinCharsAppId | out | Minimum number of characters for AppId | ui2 |
| MaxCharsAppId | out | Maximum number of characters for AppId | ui2 |
| AllowedCharsAppId | out | Allowed characters for AppId | string |
| MinCharsAppDisplayName | out | Minimum number of characters for AppDisplayName | ui2 |
| MaxCharsAppDisplayName | out | Maximum number of characters for AppDisplayName | ui2 |
| MinCharsAppUsername | out | Minimum number of characters for AppUsername | ui2 |
| MaxCharsAppUsername | out | Maximum number of characters for AppUsername | ui2 |
| AllowedCharsAppUsername | out | Allowed characters for AppUsername | string |
| MinCharsAppPassword | out | Minimum number of characters for AppPassword | ui2 |
| MaxCharsAppPassword | out | Maximum number of characters for AppPassword | ui2 |
| AllowedCharsAppPassword | out | Allowed characters for AppPassword | string |
| MinCharsIPSecIdentifier | out | Minimum number of characters for IPSecIdentifier | ui2 |
| MaxCharsIPSecIdentifier | out | Maximum number of characters for IPSecIdentifier | ui2 |
| AllowedCharsIPSecIdentifier | out | Allowed characters for IPSecIdentifier | string |
| MinCharsIPSecPreSharedKey | out | Minimum number of characters for IPSecPreSharedKey | ui2 |
| MaxCharsIPSecPreSharedKey | out | Maximum number of characters for IPSecPreSharedKey | ui2 |
| AllowedCharsIPSecPreSharedKey | out | Allowed characters for IPSecPreSharedKey | string |
| MinCharsIPSecXauthUsername | out | Minimum number of characters for IPSecXauthUsername | ui2 |
| MaxCharsIPSecXauthUsername | out | Maximum number of characters for IPSecXauthUsername | ui2 |
| AllowedCharsIPSecXauthUsername | out | Allowed characters for IPSecXauthUsername | string |
| MinCharsIPSecXauthPassword | out | Minimum number of characters for IPSecXauthPassword | ui2 |
| MaxCharsIPSecXauthPassword | out | Maximum number of characters for IPSecXauthPassword | ui2 |
| AllowedCharsIPSecXauthPassword | out | Allowed characters for IPSecXauthPassword | string |
| AllowedCharsCryptAlgos | out | Allowed characters for CryptAlgos | string |
| AllowedCharsAppAVMAddress | out | Allowed characters for AppAVMAddress | String |

| Argument name | Direction | Meaning | Data type |
|--------------------|-----------|--|-----------|
| MinCharsFilter | out | Minimum number of characters for Filter. | ui2 |
| MaxCharsFilter | out | Maximum number of characters for Filter. | ui2 |
| AllowedCharsFilter | out | Allowed characters for Filter. | string |

Table 7: Argument list of action GetInfo

3.5 RegisterApp

This action registers a new app instance. A new app instance is created if an app instance with identifier AppId does not already exists. Otherwise the existing app instance is overwritten (any VoIP and VPN configuration of this app instance is deleted).

- Required rights: C or P or N or H (only allowed from home network)
- The security context of this action must belong to a box user (no app instance allowed).
- In case AppInternetRights is true and the app instance will get access rights from the internet and new port forwardings must be (automatically) activated (e.g. for HTTPS and FTP) configuration rights (write access) are required.
- This action must be called from within the home network (from internet not allowed).
- The effective access rights of the app instance are determined by the intersection of these rights and the rights of the box user of the security context this action was called in. Specific NAS directory access rights are not included in this parameter but determined by the box user only.

| Argument name | Direction | Meaning | Data type |
|----------------|-----------|---|----------------------|
| AppId | in | Unique identifier of the app instance (unique within the single box). | string |
| AppDisplayName | in | User friendly display name of the app instance. | string |
| AppDeviceMAC | in | MAC address of the device (or device interface). Empty string means: unknown | string (MAC address) |
| AppUsername | in | Username for the app instance. This must be unique (case insensitive) among all usernames used for authentication at the box (e.g. name and email addresses of box users and usernames of other app instances). Allowed characters: a-z, A-Z, 0-9 Must not begin with a digit. Length: 1-32 characters | string |
| AppPassword | in | Password for the app instance. Username and password can be used by the app as credentials to authenticate at the following services of the box: HTTPS, TR-064, FTPS. Allowed characters: ASCII 32 – ASCII 126 Length: 8-32 characters Strength: must contain at least one digit, one uppercase letter, one lowercase letter and one other (special) character. OPEN | string |
| AppRight | in | FRITZ!OS app specific configuration right. | RightEnum |
| NasRight | in | FRITZ!OS NAS specific right. | RightEnum |

| Argument name | Direction | Meaning | Data type |
|--------------------|-----------|--|-----------|
| PhoneRight | in | FRITZ!OS phone specific right. | RightEnum |
| HomeautoRight | in | FRITZ!OS home automation specific right. | RightEnum |
| ApplInternetRights | in | Does the app instance want access the box from the internet? | boolean |

Table 8: Argument list of action RegisterApp

| Return code | Description | Related argument |
|-------------|--|---|
| 402 | Invalid arguments | Any |
| 606 | Action not authorized | |
| 803 | Argument contains invalid characters | Any input parameter |
| 820 | Internal error | |
| 821 | AppID is too short | NewAppId |
| 822 | AppID is too long | NewAppId |
| 823 | AppID contains invalid characters | NewAppId |
| 824 | AppDisplayName is too short | NewAppDisplayName |
| 825 | AppDisplayName is too long | NewAppDisplayName |
| 826 | AppDisplayName contains invalid characters | NewAppDisplayName |
| 827 | Invalid MAC-Address | NewAppDeviceMac |
| 828 | AppUserName is too short | NewAppUsername |
| 829 | AppUserName is too long | NewAppUsername |
| 830 | AppUserName contains invalid characters | NewAppUsername |
| 831 | AppUserName already exists | NewAppUsername |
| 832 | AppPassword is too short | NewAppPassword |
| 833 | AppPassword is too long | NewAppPassword |
| 834 | AppPassword contains invalid characters | NewAppPassword |
| 835 | Maximum numbers of apps has been reached | |
| 836 | No allowed right value | NewAppRight NewNasRight NewPhoneRight NewHomeautoRight NewAppInternetRights |
| 837 | Security context | |
| 838 | App may not register app | |
| 839 | App may not register from internet | |
| 840 | App no rights | |
| 841 | Enable HTTPS failed by register app Does not have to be interpreted as an error. The app instance is created. But HTTPS is inactive. | |

Table 9: Return codes of action RegisterApp

3.6 ResetEvent

Reset an event specified by a event ID. If more than one event with the same ID exist, only one event is reset.

- Required rights: configuration
- This action must be called from within the home network (from internet not allowed).

| Argument name | Direction | Meaning | Data type |
|---------------|-----------|---------|-----------|
| EventId | in | | ui4 |

Table 10: Argument list of action ResetEvent

| Return code | Description | Related argument |
|-------------|--------------------------------------|------------------|
| 402 | Invalid arguments | Any |
| 603 | Device is out of memory | Any |
| 606 | Unauthorized. Wrong security context | |
| 501 | Event with EventId not found | EventId |
| 820 | Internal error | |

Table 11: Return codes of action ResetEvent

3.7 SetAppMessageFilter

Multiple message filters of different types can be added to each app instance. If an app instance already contains a filter with the same type, it is overwritten. A filter consists of the type and a list of filter criteria. If an empty list of filter criteria is passed, the filter of this type is completely removed from the app configuration.

So far, the following filter criteria are specified:

- | | |
|---------|---|
| Type: | aha_ident (blacklist filter) |
| Filter: | Comma-separated list of AHA identifiers (eg. ain, aha_group_id, mac or vendor specific identifier) |
| Type: | tel_local_number (blacklist filter) |
| Filter: | Comma-separated list of phone numbers |

- Required rights: any
- The security context of this action **must belong to an app instance**. The app instance of the security context must be the same as the one identified by the parameter AppId.

| Argument name | Direction | Meaning | Data type |
|---------------|-----------|---|------------------|
| AppId | in | Identifier of the app instance the message filter belongs to. | string |
| Type | in | Type of the message filter, eg. "aha_ident" | string |
| Filter | in | A stringlist of filter criteria (e.g. "08761 0000444,34:45:12:43:55" for type aha_ident) Allowed characters: a-Z,0-9, Space and ,+:-_ | string (1024) |

Table 12: Argument list of action SetAppMessageFilter

| Return code | Description | Related argument |
|-------------|--------------------------------------|------------------|
| 402 | Invalid arguments | Any |
| 603 | Device is out of memory | Any |
| 606 | Unauthorized. Wrong security context | AppId |
| 714 | App with AppId not found | AppId |
| 801 | String too short | AppId |
| 802 | String too long | AppId, Filter |
| 803 | Argument contains invalid characters | AppId, Filter |
| 820 | Internal error | |

Table 13: Return codes of action SetAppMessageFilter

3.8 SetAppMessageReceiver

Configuration of a message receiver for the app instance. Every app instance can have at most only one message receiver configuration. In case a message receiver already exists for the app instance, the old configuration is overwritten (deleted).

- Required rights: any
- The security context of this action **must belong to an app instance**. The app instance of the security context must be the same as the one identified by the parameter AppId.
- This action must be called from within the home network (from internet not allowed).

| Argument name | Direction | Meaning | Data type |
|--------------------|-----------|---|-----------|
| AppId | in | Identifier of the app instance the message belongs to. | string |
| CryptAlgos | in | Comma separated list of additional crypt algorithms the app understands beside AES128-CBC-HMAC-SHA-256. If no other crypt algorithms are supported this parameter can be left blank. Naming according to RFC7518 (JWA). | string |
| AppAVMAddress | in | “App-AVM-Address” of the app Instance. An empty string means that the app instance will no longer receive any messages from this box (message receiver delete operation). The app gets this value from the AVM message relay web service. | string |
| AppAVMPasswordHash | in | BASE64URL encoding (without padding) of first 16 Bytes of the SHA-256 hash of the app's “App-AVM-Password”. The app gets this value from the AVM message relay web service. | string |
| EncryptionSecret | out | Shared secret used to build the crypt key for encryption and authentication for messages from the box to the app. | string |
| BoxSenderId | out | Sender Id used in messages from this box to the app. The app can associate this BoxSenderId to the EncryptionKey to find the needed key to decrypt a received message from a box. | string |

Table 14: Argument list of action SetAppMessageReceiver

| Return code | Description | Related argument |
|-------------|--------------------------------------|----------------------------------|
| 402 | Invalid arguments | Any |
| 501 | Action failed | |
| 603 | Device is out of memory | Any |
| 606 | Unauthorized. Wrong security context | AppId |
| 714 | App with AppId not found | AppId |
| 801 | String too short | AppId |
| 802 | String too long | AppId, CryptAlgos, AppAVMAddress |
| 803 | Argument contains invalid characters | AppId, CryptAlgos, AppAVMAddress |
| 820 | Internal error | |

Table 15: Return codes of action SetAppMessageReceiver

3.9 SetAppVPN

Configuration of a VPN (IPsec) access for the app instance. Every app instance can have at most only one VPN access configuration. In case a VPN access already exists for the app instance, the old configuration is overwritten. In case a VPN access already exists and all IPsec parameter are empty, the existing VPN configuration will be deleted.

- Required rights: configuration (write access).
- NOTE that an app instance never has any configuration rights, so **box user credentials must be used** for TR-064 authentication to perform this action.

| Argument name | Direction | Meaning | Data type |
|--------------------|-----------|--|-----------|
| AppId | in | Identifier of the app instance where the VPN configuration belongs to. | string |
| IPSecIdentifier | in | IPSec identifier | string |
| IPSecPreSharedKey | in | IPSec pre-shared-key | string |
| IPSecXauthUsername | in | Username for xauth | string |
| IPSecXauthPassword | in | Password for xauth | string |

Table 16: Argument list of action SetAppVPN

| Return code | Description | Related argument |
|-------------|--------------------------------------|---------------------|
| 402 | Invalid arguments | Any |
| 714 | App with AppId not found | AppId |
| 801 | String too short | Any input parameter |
| 802 | String too long | Any input parameter |
| 803 | Argument contains invalid characters | Any input parameter |
| 820 | Internal error | |

Table 17: Return codes of action SetAppVPN

3.10 SetAppVPNwithPFS

Configuration of a VPN (IPsec) access for the app instance with PFS (Perfect Forward Secrecy) support. Every app instance can have at most only one VPN access configuration. In case a VPN access already exists for the app instance, the old configuration is overwritten. In case a VPN access already exists and all IPSec parameter are empty, the existing VPN configuration will be deleted.

- Required rights: configuration (write access).
- NOTE that an app instance never has any configuration rights, so **box user credentials must be used** for TR-064 authentication to perform this action.

| Argument name | Direction | Meaning | Data type |
|--------------------|-----------|--|-----------|
| AppId | in | Identifier of the app instance where the VPN configuration belongs to. | string |
| IPSecIdentifier | in | IPSec identifier | string |
| IPSecPreSharedKey | in | IPSec pre-shared-key | string |
| IPSecXauthUsername | in | Username for xauth | string |
| IPSecXauthPassword | in | Password for xauth | string |

Table 18: Argument list of action SetAppVPNwithPFS

| Return code | Description | Related argument |
|-------------|--------------------------------------|---------------------|
| 402 | Invalid arguments | Any |
| 714 | App with AppId not found | AppId |
| 801 | String too short | Any input parameter |
| 802 | String too long | Any input parameter |
| 803 | Argument contains invalid characters | Any input parameter |
| 820 | Internal error | |

Table 19: Return codes of action SetAppVPNwithPFS

3.11 GetBoxSenderId

Return the box sender id.

- Required rights: any
- The security context of this action **must belong to an app instance**.

| Argument name | Direction | Meaning | Data type |
|---------------|-----------|---|-----------|
| AppId | in | Identifier of the app instance the message belongs to. | string |
| BoxSenderId | out | Sender Id used in messages from this box to the app. The app can associate this BoxSenderId to the EncryptionKey to find the needed key to decrypt a received message from a box. | string |

Table 20: Argument list of action GetBoxSenderId

| Return code | Description | Related argument |
|-------------|--------------------------------------|------------------|
| 402 | Invalid arguments | Any |
| 606 | Unauthorized. Wrong security context | AppId |
| 714 | App with AppId not found | AppId |
| 820 | Internal error | |

Table 21: Return codes of action GetBoxSenderId

4 Parameters

4.1 RightEnum

| Value | Description |
|-----------|----------------------------|
| RW | Read and write access |
| RO | Read only access |
| NO | No access allowed |
| UNDEFINED | No specific right defined. |

Table 22: Parameter type RightEnum

4.2 EventId

| Value | Description |
|-------|---------------------|
| 300 | JASON message |
| 500 | SIP register failed |
| 501 | SIP failure |
| 502 | Possible fraud |

Table 23: Parameter type EventId

5 Service States Table

| Variable name | Allowed values (* == default) | Data type |
|--------------------------------|--|----------------------|
| AccessFromInternet | 0 | Boolean |
| AllowedCharsAppAVMAddress | 0123456789ABCDEFHIJKLMNOPQRS TUVWXYZabcdefghijklmnopqrstuvwxyz-_-` | string |
| AllowedCharsAppId | 0123456789 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz | string |
| AllowedCharsAppPassword | 0123456789 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz !"#\$%&()'-.:/;<>?@[{}^`]{}` | string |
| AllowedCharsAppUsername | 0123456789 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz | string |
| AllowedCharsCryptAlgos | 0123456789ABCDEFHIJKLMNOPQRS TUVWXYZabcdefghijklmnopqrstuvwxyz-_-` | string |
| AllowedCharsFilter | 0123456789ABCDEFGHIJKLMNOPQRS TUVWXYZabcdefghijklmnopqrstuvwxyz, +:-_<SPACE> | string |
| AllowedCharsIPSecIdentifier | 0123456789 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz | string |
| AllowedCharsIPSecPreSharedKey | 0123456789 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz | string |
| AllowedCharsIPSecXauthPassword | 0123456789 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz | string |
| AllowedCharsIPSecXauthUsername | 0123456789 ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz | string |
| AppAVMAddress | | string |
| AppAVMPasswordHash | | string |
| AppDeviceMAC | | string (MAC address) |
| AppDisplayName | | string |
| AppId | | string |
| AppInternetRights | 0 | Boolean |
| AppPassword | | string |
| AppRight | NO, RO, RW, UNDEFINED | RightEnum |
| AppUsername | | string |
| BoxSenderId | | string |
| ConfigRight | NO, RO, RW, UNDEFINED | RightEnum |

| Variable name | Allowed values (* == default) | Data type |
|----------------------------|-------------------------------|-----------|
| CryptAlgos | | string |
| DialRight | NO, RO, RW, UNDEFINED | RightEnum |
| EncryptionSecret | | string |
| EventId | 0 | ui4 |
| ExternalIPAddress | | string |
| ExternalIPv6Address | | string |
| Filter | a-Z,0-9, space and ,+:-_ | string |
| FilterList | | string |
| HomeautoRight | NO, RO, RW, UNDEFINED | RightEnum |
| InternetRights | 0 | Boolean |
| IPAddress | | string |
| IPSecIdentifier | | string |
| IPSecPreSharedKey | | string |
| IPSecXauthPassword | | string |
| IPSecXauthUsername | | string |
| MaxCharsAppDisplayName | 256 | ui2 |
| MaxCharsAppId | 256 | ui2 |
| MaxCharsAppPassword | 32 | ui2 |
| MaxCharsAppUsername | 32 | ui2 |
| MaxCharsFilter | 1024 | ui2 |
| MaxCharsIPSecIdentifier | 256 | ui2 |
| MaxCharsIPSecPreSharedKey | 16 | ui2 |
| MaxCharsIPSecXauthPassword | 128 | ui2 |
| MaxCharsIPSecXauthUsername | 256 | ui2 |
| MinCharsAppDisplayName | 1 | ui2 |
| MinCharsAppId | 1 | ui2 |
| MinCharsAppPassword | 8 | ui2 |
| MinCharsAppUsername | 1 | ui2 |
| MinCharsFilter | 0 | ui2 |
| MinCharsIPSecIdentifier | 1 | ui2 |
| MinCharsIPSecPreSharedKey | 1 | ui2 |
| MinCharsIPSecXauthPassword | 1 | ui2 |
| MinCharsIPSecXauthUsername | 1 | ui2 |
| MyFritzDynDNSEnabled | 0* | Boolean |
| MyFritzDynDNSDomain | | string |
| NasRight | NO, RO, RW, UNDEFINED | RightEnum |
| PhoneRight | NO, RO, RW, UNDEFINED | RightEnum |
| RemoteAccessDDNSEnabled | 0* | Boolean |

| Variable name | Allowed values (* == default) | Data type |
|------------------------|--|-----------|
| RemoteAccessDDNSDomain | | string |
| SubnetMask | | string |
| Type | Allowed values: aha_ident, tel_local_number | string |

Table 24: Variable list