

Sofern Du VIOLET in Deine Hausautomation einbindest, konfiguriere ihre Netzwerkeinstellungen so, wie es für Server, die eine permanente Erreichbarkeit in einem Netzwerk haben sollen, üblich ist! Richte **eine feste IP-Adresse** in VIOLETs Netzwerkeinstellungen ein (**außerhalb des DHCP Bereiches Deines Routers**) und richte alle Abfragen **direkt an diese IP-Adresse** und **nicht** an die local Domain (`http://violet.local`) um

1. sicherzustellen, dass Deine Hausautomation VIOLET immer unter der gleichen Adresse erreicht
2. um unnötige und typischerweise langsame Namesauflösungen in Deinem Netzwerk zu vermeiden.

26.1 JSON-API, ABFRAGE VON MESSWERTEN

VIOLET kann alle Messwerte über eine JSON API im Netzwerk zur Verfügung stellen, damit Du Deine Pooldaten auch in jeder beliebigen Hausautomation (KNX, Loxone, IP-Symcon, IO-Broker, etc...) mit visualisieren kannst. Du kannst Dir die Ausgabe der Daten z.B. im Browser durch Aufruf der URL `http://violet.local/getReadings?ALL` ansehen.

Die API kann Dir pauschal alle Messwerte / Zustände liefern oder Du frägst nur einzelne Daten oder ganze Gruppen von Messwerten ab. Ein GET Request in folgendem Format liefert z.B. alle maximal möglichen Werte incl. der Stati aller Ausgänge und Digital-Inputs zurück:

`/getReadings?ALL`

Die Beschreibung aller Werte ist in einer separaten Excel-Liste als Download verfügbar:

https://www.myViolet.de/_violet/paperwork/api_description/getReadings.xlsx

ABFRAGE EINZELNER WERTE (Beispiele):

GET Request: `/getReadings?pH_value`

(Liefert den aktuellen pH Messwert und das Tages-Minimum und Maximum – also jeden Messwert, der im Namen *`pH_value`* enthält)

Response:

```
{  
  "pH_value_min": 7.22,  
  "pH_value_max": 7.30,  
  "pH_value": 7.30,  
}
```

GET Request: `/getReadings?onewire1_value`

Liefert den aktuellen Messwert des 1. Temperatursensors (Wassertemperatur)

Response:

```
{  
  "onewire1_value": 28.1,  
  "onewire1_value_min": 27.9,
```

```
"onewire1_value_max": 28.8  
}
```

GET Request: `/getReadings?onewire`

Liefert alle Daten zurück, die „onewire“ enthalten

Response (verkürzte Darstellung):

```
{  
  "onewire1_value":28.1,"onewire2_value":14.7,"onewire3_value":22.8,  
  ...  
  "onewire1_value_min":"26.2","onewire1_value_max":"28.8","onewire2_value_min":"14.2",  
  ...  
  "onewire1_faultcount":0,"onewire2_faultcount":0,"onewire3_faultcount":0,  
  ...  
  "onewire1_state":"OK","onewire1_rcode":"28121883321901A9",  
  "onewire2_state":"OK","onewire2_rcode":"28169E4C32190181",  
  "onewire3_state":"OK","onewire3_rcode":"2811FD713219018E  
  ...  
  "onewire10_rcode":"0","onewire10_state":"NO_SENSOR_CONFIGURED",  
  "onewire11_rcode":"0","onewire11_state":"NO_SENSOR_CONFIGURED  
}
```

Der Query-String kann auch eine beliebige Anzahl gewünschter Werten enthalten die durch „,“ (Komma) getrennt werden (z.B. `getReadings?pH_value,orp_value,pot_value`) oder nur einen Teil eines Wertennames (z.B. `getReadings?_value`)

GET Request: `/getReadings?pH_value,orp_value,pot_value`

Liefert alle pH-, ORP- (Redox) und Potentiostat (Chlor) Messwerte zurück

Response:

```
{  
  "pH_value_min": 7.22,  
  "pH_value_max": 7.30,  
  "pH_value": 7.30,  
  "orp_value_min": 722.4,  
  "orp_value_max": 781.0,  
  "orp_value": 775.4,  
  "pot_value_min": 0.12,  
  "pot_value_max": 0.66,  
  "pot_value": 0.32  
}
```

GET Request: `/getReadings?_value`

Liefert jeden Wert zurück der `_value` enthält

Response (verkürzte Ansicht)

```
{  
  "onewire1_value": 27.8,
```

```

"onewire2_value": 14.4,
"onewire3_value": 22.6,
...
"onewire1_value_min":"0.0",
"onewire1_value_max":"28.2",
"onewire2_value_min":"0.0",
...
"pH_value_min":"7.22",
"pH_value_max":"7.30",
"orp_value_min":"722.4",
"orp_value_max":"787.1",
"pot_value_min":"0.12",
"pot_value_max":"0.66",
"IMP1_value":"12.4",           /* Durchfluss Messwasser */
"IMP2_value":"0.00",           /* Impulseingang 2. */
"ADC1_value":"0.45",           /* Analogsensor 1 (Filterdruck) */
"ADC2_value":"49.4",           /* Analogsensor 2 (Pegelstand) */
...
"pH_value":"7.30",
"orp_value":"775.4",
"pot_value":"0.32"
}

```

GET Request: */getReadings?DOSING*

Liefert die Tagesdosiermengen zurück

Response:

```
{
"DOS_1_CL_DAILY_DOSING_AMOUNT_ML":"2204",           /* Tages-Produktionsleistung in Gramm */
"DOS_2_ELO_DAILY_DOSING_AMOUNT_ML":"0.0",
"DOS_4_PHM_DAILY_DOSING_AMOUNT_ML":"144",
"DOS_5_PHP_DAILY_DOSING_AMOUNT_ML":"0",
"DOS_6_FLOC_DAILY_DOSING_AMOUNT_ML":"84"
}
```

Die Beschreibung aller Werte ist in einer separaten Excel-Datei als Download verfügbar:

https://www.myViolet.de/_violet/paperwork/api_description/getReadings.xlsx

26.2 AUSLÖSEN VON STEUERUNGSOPTIONEN PER NETZWERKREQUEST

Einige Steuerungsfunktionen, die Du über VIOLET's Dashboard auslösen kannst, kannst Du ebenfalls über Deine Hausautomation auslösen (z.B. manuelles Schalten der Filterpumpe oder der Beleuchtung / DMX-Szenen). Die konfigurierten Schaltregeln für die Schaltiereingänge und die Relais auf den Relaiserweiterungen, lassen sich ebenfalls per Netzwerk-Request auslösen.

Der Request, den Du dazu an VIOLET senden musst, ist ein einfacher GET Request. Er muss im Header die Authentifizierung enthalten (BasicAUTH) und an `.../setFunctionManually?` gesendet werden. Er muss vier weitere Parameter im Query enthalten (durch Kommas getrennt):

`.../setFunctionManually?{AUSGANG},{SCHALTZUSTAND},{WERT_1},{WERT_2}`

Die Groß- und Kleinschreibung der Parameter ist zu beachten (siehe Beispiele).

Sofern Du VIOLET in Deine Hausautomation einbindest, konfiguriere ihre Netzwerkeinstellungen so, wie es für Server, die eine permanente Erreichbarkeit in einem Netzwerk haben sollen, üblich ist! Richte **eine feste IP-Adresse** in VIOLET's Netzwerkeinstellungen ein (**außerhalb des DHCP Bereiches Deines Routers**) und richte alle Abfragen **direkt an diese IP-Adresse** und nicht an die local Domain (`http://violet.local`) um

1. sicherzustellen, dass Deine Hausautomation VIOLET immer unter der gleichen Adresse erreicht
 2. unnötige und typischerweise langsame Namesauflösungen in Deinem Netzwerk zu vermeiden.
-

JEDEN AUSGANG, DEN DU ÜBER DIESE FUNKTION AKTIVIERST ODER DEAKTIVIERTS MUSST DU AUCH SELBST WIEDER IN DEN GEWÜNSCHTEN STATUS ZURÜCKSETZEN! Verwendest Du die Zeitfunktionen nicht, bleibt der Ausgang ansonsten permanent in dem Status, in den Du ihn gesetzt hast (ON | OFF).

Die Schaltbefehle, die Du über diese Requests auslöst, werden wie ein manuelles EIN oder ein manuelles AUS priorisiert -> Manuelles AUS bleibt also permanent AUS, manuelles EIN wird ggf. nur von Sicherheitsfunktionen (z.B. Trockenlaufschutz oder Drucküberwachung der Filterpumpe) übersteuert.

Bei Deiner eigenen Umsetzung musst Du entsprechende Vorkehrungen treffen, mit denen Du sicherstellen kannst, dass die Requests Deiner Hausautomation auch im Falle kurzfristiger Störungen oder Netzwerkunterbrechungen (innerhalb Deines Netzwerkes) bei VIOLET ankommen bzw. entsprechend wiederholt werden!

26.2.1 OPTIONEN FÜR DIE FILTERPUMPE

{AUSGANG}	{SCHALTZUSTAND}	{WERT_1}	{WERT_2}
PUMP	ON AUTO OFF	Laufzeit in Sekunden, die der entsprechende Schaltzustand aktiv bleiben soll (0 für permanent)	Drehzahl für die Filterpumpe, sofern eine regelbare Pumpe angeschlossen ist.

Beispiele:

<http://violet.local/setFunctionManually?PUMP,ON,120,2>

...schaltet den Ausgang PUMP für zwei Minuten (120 Sekunden) auf ON. Sofern eine drehzahlgeregelte Pumpe angeschlossen ist, wird diese auf Drehzahl_2 geschaltet. Die Pumpe geht nach Ablauf der Zeit in den Zustand AUTO.

<http://violet.local/setFunctionManually?PUMP,ON,0,2>

...schaltet den Ausgang PUMP permanent auf ON. Sofern eine drehzahlgeregelte Pumpe angeschlossen ist, wird diese auf Drehzahl_2 geschaltet. Die Pumpe bleibt in diesem Zustand.

<http://violet.local/setFunctionManually?PUMP,AUTO,0,0>

...setzt die Filterpumpe zurück auf AUTO, eine mitübergebene Laufzeit oder Drehzahl wird nicht weiter ausgewertet.

<http://violet.local/setFunctionManually?PUMP,OFF,600,0>

...setzt die Filterpumpe für 10 Minuten (600 Sekunden) zwangsweise auf AUS, eine mitübergebene Drehzahl in {WERT_2} wird nicht weiter ausgewertet. Die Pumpe geht nach Ablauf der Zeit in den Zustand AUTO.

Der Response Body auf diese Requests enthält bis zu vier Zeilen (text/plain), die Du entsprechend auswerten solltest um im Fehlerfall (Übertragungsfehler, Netzwerkunterbrechung, Fehler in der Antwort) den Request auf jedenfall wiederholen zu können. Nur so ist sichergestellt, dass die Funktion nicht unerwünscht / dauerhaft in Betrieb bleibt.

Beispiel:

Name	X	Header	Nutzlast	Vorschau	Antwort	Initiator	Timing
setFunctionManually?PUMP,ON,900,2					1 OK 2 PUMP 3 SWITCHED_TO_ON 4 TILL 16.09. 23:24		

- 1. Zeile: OK oder ERROR. Sollte zur Auswertung herangezogen werden.
- 2. Zeile: {OUTPUT} der entsprechende Ausgang der geschaltet werden sollte.
- 3. Zeile: {TEXT} Info zum Schaltzustand.
- 4. Zeile: {TEXT} Ggf. zusätzliche Informationen

26.2.2 OPTIONEN FÜR DIE RELAIS AUF DER/DEN RELAISERWEITERUNG/EN

{AUSGANG}	{SCHALTZUSTAND}	{WERT_1}	{WERT_2}
EXT1_1 EXT1_2 ... EXT1_8	ON AUTO OFF	Laufzeit in Sekunden, die der entsprechende Schaltzustand aktiv bleiben soll (0 für permanent).	Nicht verwendet
EXT2_1 EXT2_1 ... EXT2_8	ON AUTO OFF	Laufzeit in Sekunden, die der entsprechende Schaltzustand aktiv bleiben soll (0 für permanent)	Nicht verwendet

Beispiele:

http://violet.local/setFunctionManually?EXT1_3,ON,3600,0

...schaltet Relais_3 auf der 1. Relaiserweiterung eine Stunde (3600 Sekunden) auf EIN. Das Relais fällt nach Ablauf der Zeit in den Zustand AUTO zurück.

http://violet.local/setFunctionManually?EXT1_5,OFF,120,0

...schaltet Relais_5 auf der 1. Relaiserweiterung für 120 Sekunden auf MANUELL AUS. Das Relais fällt nach Ablauf der Zeit in den Zustand AUTO zurück.

Die Relais _6, _7, _8 auf der 1. Relaiserweiterung kannst Du auf diesem Weg nicht aktivieren, sofern Du die Cover-Steuerung verwendest. Sie sind für diese und ähnliche Anwendungen gesperrt und können nur über das Dashboard (Cover) bedient werden.

26.2.3 OPTIONEN FÜR DEN AUSGANG LIGHT UND DIE DMX-SZENEN

{AUSGANG}	{SCHALTZUSTAND}	{WERT_1}	{WERT_2}
LIGHT	ON AUTO OFF COLOR	Nicht verwendet, keine Zeitfunktion verfügbar COLOR schaltet den Ausgang LIGHT für etwa 150ms ab und wieder an	Nicht verwendet
DMX_SCENE1 DMX_SCENE2 ... DMX_SCENE12	ON AUTO OFF	Nicht verwendet, keine Zeitfunktion verfügbar	Nicht verwendet
Beliebige Szene	ALLON ALLAUTO ALLOFF	Setze alle DMX-Szenen und den Ausgang LIGHT gleichzeitig auf ON, AUTO oder OFF	

Beispiele:

<http://violet.local/setFunctionManually?LIGHT,ON,0,0>

...schaltet den Ausgang LIGHT EIN. Eine mitübergebene Laufzeit wird hier nicht ausgewertet. Es steht für die Beleuchtung keine Zeitfunktion zur Verfügung.

<http://violet.local/setFunctionManually?LIGHT,COLOR,0,0>

...schaltet den Ausgang LIGHT für etwa 150ms AUS (Farbwechsel für einfache LED-Leuchten) und danach wieder ein (sofern der Ausgang vorher eingeschaltet war).

http://violet.local/setFunctionManually?DMX_SCENE3,ON,0,0

...schaltet DMX_SCENE3 EIN. Eventuell in der Konfiguration verknüpfte Szenen werden dabei automatisch auf manuell AUS gesetzt.

http://violet.local/setFunctionManually?DMX_SCENE1,ALLAUTO,0,0 oder z.B.

http://violet.local/setFunctionManually?DMX_SCENE11,ALLAUTO,0,0

...schaltet alle DMX-Szenen und den Ausgang LIGHT zurück auf AUTO. Es ist unerheblich, welche Szene für den Wert {AUSGANG} übergeben wird.

http://violet.local/setFunctionManually?DMX_SCENE1,ALLON,0,0

...schaltet alle DMX-Szenen und den Ausgang LIGHT manuell EIN. Es ist unerheblich, welche Szene für den Wert {AUSGANG} übergeben wird.

26.2.4 OPTIONEN FÜR DIE SCHALTREGELN / SCHALTEREINGÄNGE

{AUSGANG}	{SCHALTZUSTAND}	{WERT_1}	{WERT_2}
DIRULE_1	PUSH	Nicht verwendet, keine Zeitfunktion verfügbar.	
DIRULE_2	LOCK	Zeiten werden beim Auslösen einer Schaltregel aus der Konfiguration der Schaltregel übernommen.	
...			
DIRULE_7	UNLOCK		Nicht verwendet

Beispiele:

http://violet.local/setFunctionManually?DIRULE_1,PUSH,0,0

...löst die konfigurierte Schaltregel Nummer_1 aus (gleiche Funktion wie das Auslösen über das Dashboard oder einen an den (konfigurierten) Digital-Input angeschlossenen Taster).

http://violet.local/setFunctionManually?DIRULE_3,LOCK,0,0

...sperrt die konfigurierte Schaltregel Nummer 3. Ein Auslösen der Schaltregel, durch eventuell angeschlossene Taster, ist nicht mehr möglich.

http://violet.local/setFunctionManually?DIRULE_3,UNLOCK,0,0

...entsperrt die konfigurierte Schaltregel Nummer 3.

Ist für eine Schaltregel – die über einen Netzwerk-Request ausgelöst wird - die Auslöseart „NORMAL“ (Schalter) konfiguriert, wird die Regel für etwa 500ms ausgelöst und wieder gestoppt

26.3 PHOTOVOLTAIK-ÜBERSCHUSS PER NETZWERK AUSLÖSEN

Die Steuerungsfunktion Photovoltaik-Überschuss (für Filterpumpe und Wärmepumpe konfigurierbar) kannst Du entweder durch einen potentialfreien Relaiskontakt (der an VIOLET's „PVS“ Eingang angeschlossen wird) auslösen oder per HTTP (GET) Request, durch ein übergeordnetes Energiemanagement / Hausautomations-System.

Im Falle einer Auslösung per HTTP Request bleibt die Betriebsart PV-Überschuss solange aktiviert, bis sie wieder durch einen HTTP Request zurückgenommen wird. Bei Deiner eigenen Umsetzung musst Du also entsprechende Vorkehrungen treffen mit denen Du sicherstellen kannst, dass die Requests Deines Energiemanagements oder Deiner Hausautomation auch im Falle kurzfristiger Netzwerkunterbrechungen / Störungen bei VIOLET ankommen bzw. entsprechend wiederholt werden.

Sofern Du diese Funktion nutzt oder VIOLET's Daten (z.B. über eine Hausautomation) regelmäßig abfrägst, vergib Violet immer eine festen IP-Adresse und richte die Abfragen immer direkt an diese IP-Adresse - nicht an den lokalen Domainnamen (`violet.local`) - um

1. sicherzustellen, dass Deine Hausautomation VIOLET immer unter der gleichen Adresse erreicht
 2. um unnötige und typischerweise langsame Namesauflösungen in Deinem Netzwerk zu vermeiden.
-

Der HTTP Request erfolgt ebenfalls an `/setFunctionManually`, muss im Header die Authentifizierung enthalten (BasicAUTH mit Benutzername / Kennwort) und als Query die entsprechenden Parameter.

Beispiel:

<http://violet.local/setFunctionManually?PVSURPLUS,ON,2> (Groß- und Kleinschreibung ist zu beachten, die einzelnen Parameter im QueryString werden durch Kommas getrennt).

{AUSGANG}	{SCHALTZUSTAND}	{WERT_1}	{WERT_2}
PVSURPLUS	ON OFF	1 oder 2 oder 3 Drehzahl für die Filterpumpe, sofern eine regelbare Pumpe angeschlossen ist.	Nicht verwendet

{WERT_1} gibt, bei Verwendung einer regelbaren Filterpumpe, die Drehzahlstufe vor, die angesteuert werden soll. Die über VIOLET's Benutzeroberfläche eingestellte Drehzahl für den PV-Überschuss der Filterpumpe wird in diesem Fall ignoriert. Wird keine Drehzahlstufe mit übergeben, wird die Drehzahl aus der Konfiguration übernommen.

Der Response Body auf diesen Request enthält drei Zeilen (text/plain), die Du entsprechend auswerten solltest um im Fehlerfall (Übertragungsfehler, Netzwerkunterbrechung, Fehler in der Antwort) den Request auf jedenfall wiederholen zu können. Nur so ist sichergestellt, dass die Funktion nicht unerwünscht / dauerhaft in Betrieb bleibt.

Name	X	Header	Nutzlast	Vorschau	Antwort	Initiator	Timing
setFunctionManually?PVSURPLUS,ON,2				1 OK 2 PVSURPLUS 3 ON 4			

1. Zeile: OK (oder ERROR)
2. Zeile: PVSURPLUS (wird immer zurückgeliefert)
3. Zeile: ON (oder OFF)

Ein Request an <http://violet.local/getReadings?PVSURPLUS> liefert Dir den aktuellen Status der PV-Überschuss-Funktion zurück (JSON formatiert).

Response:

```
{  
    "PVSURPLUS": "0" /* (oder 1 oder 2 ) */  
}
```

- 0** - PV-Überschuss-Funktion inaktiv
1 - PV-Überschuss-Funktion aktiv, ausgelöst durch Digital-Input
2 - PV-Überschuss-Funktion aktiv, ausgelöst durch HTTP Request

Die Beschreibung aller Werte der Abfrage an /getReadings ist in einer separaten Excel-Liste als Download verfügbar: https://www.pooldigital.de/_violet/paperwork/api_description/getReadings.xlsx