

Cloud protocol

- same payload for UDP and TCP stream
- encryption key depending of Cloud/App usage
- for unprovisioned devices:
 - while discovery: Token in plaintext in the checksum field

	Byte 0,1	Byte 2,3	Byte 4,5,6,7	Byte 8,9,A,B	Byte C,D,E,F
Header	Magic:2131	Lenght (hex)	00 00 00 00	DID (hex)	epoch (hex) (big endian)
Checksum	Md5sum[Header + Key(Cloud)/Token(App) + Data(if exists)]				
Data	Encrypted Data(if exists, e.g. if not Ping/Pong or Hello message) <ul style="list-style-type: none">• token = for cloud: key; for app: token• key = md5sum(token)• iv = md5sum(key+token)• cipher = AES(key, AES.MODE_CBC, iv, padded plaintext)				

Cloud protocol

- Data
 - JSON-formated messages
 - Packet identified by packetid
 - Structures:
 - commands: "methods" + "params"
 - responses : "results"
 - Every command/response confirmed by receiver (except otc)
- Example
 - `{'id': 136163637, 'params': {'ap': {'ssid': 'myWifi', 'bssid': 'F8:1A:67:CC:BB:AA', 'rssi': -30}, 'hw_ver': 'Linux', 'life': 82614, 'model': 'rockrobo.vacuum.v1', 'netif': {'localIp': '192.168.1.205', 'gw': '192.168.1.1', 'mask': '255.255.255.0'}, 'fw_ver': '3.3.9_003077', 'mac': '34:CE:00:AA:BB:DD', 'token': 'xxx'}, 'partner_id': '', 'method': '_otc.info'}`

Cloud protocol

- Initial connection (repeated every 30 minutes)
 - Device -> Cloud

```
{'id': 136163637, 'params': {'ap': {'ssid': 'dune', 'bssid': 'F8:1A:67:CC:BB:AA', 'rssi': -30}, 'hw_ver': 'Linux', 'life': 82614, 'model': 'rockrobo.vacuum.v1', 'netif': {'localip': '192.168.1.205', 'gw': '192.168.1.1', 'mask': '255.255.255.0'}, 'fw_ver': '3.3.9_003077', 'mac': '34:CE:00:AA:BB:DD', 'token': 'xxx'}, 'partner_id': '', 'method': '_otc.info'}
```
 - Cloud -> Device

```
{'id': 136163637, 'result': {'otc_list': [{'ip': '120.92.96.243', 'port': 80}, {'ip': '124.251.57.12', 'port': 80}, {'ip': '124.251.57.11', 'port': 80}], 'otc_test': {'firsttest': 441, 'list': [{'ip': '124.251.57.10', 'port': 8053}, {'ip': '120.92.96.243', 'port': 8053}, {'ip': '123.125.103.75', 'port': 8053}, {'ip': '45.124.124.107', 'port': 8053}, {'ip': '220.181.138.201', 'port': 8053}], 'interval': 1800}}}
```

Cloud protocol

- Status pull
 - Cloud -> Device
{'id': 20, 'method': 'get_status', 'from': '4', 'params': []}
 - Device -> Cloud
{'id': 20, 'result': [{'state': 8, 'msg_ver': 5, 'map_present': 0, 'error_code': 0, 'fan_power': 38, 'in_cleaning': 0, 'clean_time': 1127, 'msg_seq': 12, 'dnd_enabled': 1, 'battery': 100, 'clean_area': 23647500}]}
 - Cloud -> Device
{'id': 20, 'result': 'ok'}

Cloud protocol

- Status push (every 30 minutes)
 - Device -> Cloud

```
{'id': 1472908087, 'method': 'props', 'params': {'ota_state': 'idle'}}
{'id': 7866501, 'method': 'event.status', 'params': [{'state': 8, 'msg_ver': 5, 'map_present': 0, 'error_code': 0, 'fan_power': 38, 'in_cleaning': 0, 'clean_time': 1127, 'msg_seq': 10, 'dnd_enabled': 1, 'battery': 100, 'clean_area': 23647500}]}
```
 - Cloud -> Device

```
{'id': 1472908087, 'result': 'ok'}
{'id': 7866501, 'result': 'ok'}
```

Cloud protocol

- Keep-Alives (~every 15 seconds)
 - Device -> Cloud
 - Empty packet with current timestamp + MD5
 - Cloud -> Device
 - Empty packet with current timestamp + MD5