Web Monitoring API Specification Document

(V2.1)

Contents

1、 Log-in

Website: https://api.alphaess.com/ras/v2/Login

Remark: This access is used to log in.

Http request method: POST https://api.alphaess.com/ras/v2/Login

Request parameter:

Parameter	Type	Description	Requires
api_account	string	Api account	Yes
timestamp	long	Unix timestamp is used to confirm validity of your request. If the timespan between request timestamp and server timestamp exceeds 300 seconds, then the request will be rejected. How to calculate the unix timestamp? (DateTime.UtcNow-new DateTime(1970,1,1)).TotalSeconds	Yes
sign	string	Rule of signing: format request data (except sign) and secret key(provided by Alpha) using Key=Value, splice the keys arranged by sort ascending into a word string, and then use MD5 digest for passwords with 32-bit hex encoding. Please refer to the Signature Algorithm	Yes
username	string	Username	Yes
password	string	Password which will be transmitted after encryption(Please refer to the Encryption Algorithm)	Yes

Response (json):

Fields	Type	Description
userType	string	User Type
ReturnCode	int	Return Code
Token	String	Authentication token (the timeout period is 90 minutes)

2、 Acquire system list of different end users.

Website: https://api.alphaess.com/ras/v2/GetSystemList

Remark: This access is used to acquire system list of different end users.

Http request method: POST https://api.alphaess.com/ras/v2/GetSystemList

Request parameter:

Parameter	Type	Description	Requires
api_account	string	Api account	Yes
timestamp	long	Unix timestamp is used to confirm validity of your request. If the timespan between request timestamp and server timestamp exceeds 300 seconds, then the request will be rejected. How to calculate the unix timestamp? (DateTime.UtcNow-new DateTime(1970,1,1)).TotalSeconds	Yes

sign	string	Rule of signing: format request data (except sign) and secret key(provided by Alpha) using Key=Value, splice the keys arranged by sort ascending into a word string, and then use MD5 digest for passwords with 32-bit hex encoding. Please refer to the Signature Algorithm	Yes
Token	string	Authentication token	Yes
pageindex	int	Page Index (Default to 1)	Yes
pagesize	int	Page Size(Default to 10)	No

Response (json)：

Fields			Type	Description
ReturnCode			int	Return Code
Result				Response
|--	TotalPageCount		int	Page count
|--	PageIndex		int	Page Index
|--	TotalPageCount		int	Page count
|--	TotalCount		int	Total records
|--	PageSize		int	Page size
|--	Systems		List<Data>	System list
	|--	Sn	string	System S/N
	|--	SystemModel	string	System Model
	|--	Cobat	decimal	Battery capacity
	|--	UsableCapacity	decimal	Usable capacity
	|--	Mbat	string	Battery type
	|--	Poinv	decimal	Rated output power of the inverter
	|--	Popv	decimal	Rated installed capacity of PV
	|--	Remark	string	Remark S/N
	|--	Solution	string	Different solutions(1:AC solution,2:DC solution,3: hybrid solution)
	|--	EmsVersion	string	Ems Version
	|--	BmsVersion	string	Bms Version
	|--	InvVersion	string	Inverter Firmware Version
	|--	InvModel	string	Inverter Model
	|--	MeterModel	string	Meter Model
	|--	MeterPhase	int	Meter Phase
	|--	SetFeed	int	Feen in %
	|--	NetWorkStatus	int	Online or not?(1：online，0：offline)
	|--	State	string	System operation status
	|--	EndUser	SysUser	End user

3、Acquire energy summary of system

Website: https://api.alphaess.com/ras/v2/GetEnergySummary

Remark: This access is used to acquire energy summary of system.

Http request method: POST https://api.alphaess.com/ras/v2/GetEnergySummary

Request parameter:

Parameter	Type	Description	Requires
api_account	string	Api account	Yes
timestamp	long	Unix timestamp is used to confirm validity of your request. If the timespan between request timestamp and server timestamp exceeds 300 seconds, then the request will be rejected. How to calculate the unix timestamp? (DateTime.UtcNow-new DateTime(1970,1,1)).TotalSeconds	Yes
sign	string	Rule of signing: format request data (except sign) and secret key(provided by Alpha) using Key=Value, splice the keys arranged by sort ascending into a word string, and then use MD5 digest for passwords with 32-bit hex encoding. Please refer to the Signature Algorithm	Yes
Token	string	Authentication token	Yes
Sn	string	If S/N is entered then date of the device relating to this S/N will be provided otherwise all date of the devices of the particular user will be provided.	No
TheDate	Date	Returns data on the specified date(date format: yyyy-MM-dd)	Yes

Response (json)：

Fields		Type	Description
ReturnCode		int	Return Code
Result		List<Data>	data list
|--	Sn	string	S/N
|--	EGrid2Load	decimal	Load consumption from grid
|--	EGridCharge	decimal	Grid-charge
|--	Ebat	decimal	Battery consumption
|--	Echarge	decimal	PV charge
|--	Eeff	decimal	Self-consumption
|--	Einput	decimal	Grid consumption
|--	Eload	decimal	Load consumption
|--	Eout	decimal	Feed-in
|--	Epv2load	decimal	Load consumption from PV
|--	EpvT	decimal	PV generation
|--	EselfConsumption	decimal	Self-consumption rate
|--	EselfSufficiency	decimal	Self-sufficiency rate

4、Access to the latest system routine operating data

Website: https://api.alphaess.com/ras/v2/GetRunningData

Remark: The interface obtains the latest system routine operating data (one piece of data per 5 minutes)

Http request method: POST https://api.alphaess.com/ras/v2/GetRunningData

Request parameter:

Parameter	Type	Description	Requires
api_account	string	Api account	Yes
timestamp	long	Unix timestamp is used to confirm validity of your request. If the timespan between request timestamp and server timestamp exceeds 300 seconds, then the request will be rejected. How to calculate the unix timestamp? (DateTime.UtcNow-new DateTime(1970,1,1)).TotalSeconds	Yes
sign	string	Rule of signing: format request data (except sign) and secret key(provided by Alpha) using Key=Value, splice the keys arranged by sort ascending into a word string, and then use MD5 digest for passwords with 32-bit hex encoding. Please refer to the Signature Algorithm	Yes
Token	string	Authentication token	Yes
Sn	string	If S/N is entered then date of the device relating to this S/N will be provided otherwise all date of the devices of the particular user will be provided.	No

Response (json)：

Fields		Type	Description
ReturnCode		int	Return Code
Result		List\<Data\>	Data list
|--	Sn	string	S/N
|--	UploadTime	datetime	Data upload time
|--	Ppv1	decimal	PV1 power
|--	Ppv2	decimal	PV2 power
|--	Upv1	decimal	PV1 voltage
|--	Upv2	decimal	PV2 voltage
|--	Ua	decimal	Grid voltage 1
|--	Ub	decimal	Grid voltage 2
|--	Uc	decimal	Grid voltage 3
|--	Fac	decimal	Grid Frequency
|--	Ubus	decimal	Busbar voltage
|--	PrealL1	decimal	L1 power of inverter
|--	PrealL2	decimal	L2 power of inverter
|--	PrealL3	decimal	L2 power of inverter
|--	Tinv	decimal	Inverter core temperature
|--	PacL1	decimal	EMS control power commands 1
|--	PacL2	decimal	EMS control power commands 2
|--	PacL3	decimal	EMS control power commands 3

|--	InvWorkMode	int	Grid state: 1：wait；2：online；4：battery mode；8：bypass； 16：Fault mode；32：VF start mode
|--	EpvTotal	decimal	PV generation energy
|--	Einput	decimal	Grid consumption energy
|--	Eoutput	decimal	Feed-in energy
|--	Echarge	decimal	Battery charging energy
|--	PmeterL1	decimal	AC meter power L1
|--	PmeterL2	decimal	AC meter power L2
|--	PmeterL3	decimal	AC meter power L3
|--	PmeterDC	decimal	DC meter power
|--	Pbat	decimal	Batter power
|--	SOC	decimal	State of charge
|--	BatV	decimal	Battery voltage
|--	BatC	decimal	Battery current
|--	FlagBms	char	BMS Flag
|--	BmsWork	int	BMS State: 0:standby 1:work 2:parallel 4: maintenance
|--	Pcharge	int	BMS allowed discharging power
|--	Pdischarge	int	BMS allowed discharging power
|--	BmsRelay	nvarchar	Relay status: XXX： Represents the state of the three relays
|--	BmsNum	nvarchar	Number of battery modules
|--	VcellLow	nvarchar	battery modules ID + battery identifier+ Minimum voltage
|--	VcellHigh	nvarchar	battery modules ID + battery identifier +Maximum voltage
|--	TcellLow	nvarchar	battery modules ID + Sampling point number + Minimum temperature
|--	TcellHigh	nvarchar	battery modules ID D+ Sampling point number + Maximum temperature
|--	IdTempLover	nvarchar	Low temperature Warning battery number
|--	IdTempEover	nvarchar	High temperature Warning battery number
|--	IdTempediffe	nvarchar	Temperature difference Warning battery number
|--	IdChargcurre	nvarchar	Charging current Warning battery number
|--	IdDischcurre	nvarchar	Discharging current Warning battery number
|--	IdCellvolover	nvarchar	Single cell overvoltage Warning battery number
|--	IdCellvollower	nvarchar	Single cell undervoltage Warning battery number
|--	IdSoclower	nvarchar	SOC critical low warning battery number
|--	IdCellvoldiffe	nvarchar	Single cell voltage difference Warning battery number
|--	BatC1	decimal	Battery 1 Current
|--	BatC2	decimal	Battery 2 Current
|--	BatC3	decimal	Battery 3 Current
|--	BatC4	decimal	Battery 4 Current
|--	BatC5	decimal	Battery 5 Current
|--	BatC6	decimal	Battery 6 Current
|--	ErrInv	nvarchar	Inverter fault

|--	WarInv	nvarchar	Inverter warning
|--	ErrEms	nvarchar	EMS fault
|--	ErrBms	nvarchar	Battery fault
|--	ErrMeter	nvarchar	Meter fault
|--	ErrBackupBox	nvarchar	BackupBox fault
|--	EGridCharge	nvarchar	Gird charged energy
|--	EmsStatus	nvarchar	Ems status
|--	EDischarge	decimal	Battery discharged energy
|--	SOC1	decimal	Battery 1 SOC
|--	SOC2	decimal	Battery 2 SOC
|--	SOC3	decimal	Battery 3 SOC
|--	SOC4	decimal	Battery 4 SOC
|--	SOC5	decimal	Battery 5 SOC
|--	SOC6	decimal	Battery 6 SOC
|--	VcellLowValue	decimal	Minimum voltage of single cell
|--	VcellHighValue	decimal	Maximum voltage of single cell
|--	TcellLowValue	decimal	Minimum temperature of single cell
|--	TcellHighValue	decimal	Maximum temperature of single cell
|--	InvBatV	decimal	Inverter sampling battery voltage
|--	BmsShutdown	int	Battery shutdown fault code
|--	BmuRelay	int	Relay status of SOC
|--	BmsHardVer1	int	Hardware version of the first set of battery cabinets（4 digits for a battery, 6 batteries in total）
|--	BmsHardVer2	int	Hardware version of the first set of battery cabinets（4 digits for a battery, 6 batteries in total）
|--	BmsHardVer3	int	Hardware version of the third set of battery cabinets（4 digits for a battery, 6 batteries in total）
|--	DispatchSwitch	decimal	Dispatching enable switch
|--	Pdispatch	decimal	Dispatching power
|--	DispatchSoc	decimal	SOC of dispatching
|--	DispatchMode	decimal	Dispatching Mode

5、Access to the historical system routine operating data

Website: https://api.alphaess.com/ras/v2/GetHistoryRunningData

Remark: The interface obtains the historical system routine operating data (Get one day at a time)

Http request method: POST https://api.alphaess.com/ras/v2/GetHistoryRunningData

Request parameter:

Parameter	Type	Description	Requires
api_account	string	Api account	Yes
timestamp	long	Unix timestamp is used to confirm validity of your request. If the timespan between request timestamp and server timestamp	Yes

		exceeds 300 seconds, then the request will be rejected. How to calculate the unix timestamp? (DateTime.UtcNow-new DateTime(1970,1,1)).TotalSeconds	
sign	string	Rule of signing: format request data (except sign) and secret key(provided by Alpha) using Key=Value, splice the keys arranged by sort ascending into a word string, and then use MD5 digest for passwords with 32-bit hex encoding. Please refer to the Signature Algorithm	Yes
Token	string	Authentication token	Yes
Sn	string	If S/N is entered then date of the device relating to this S/N will be provided otherwise all date of the devices of the particular user will be provided.	No
Starttime	string	Historical data start time (Format: yyyy/MM/dd HH:mm:ss)	Yes
endtime	string	Historical data end time (Format: yyyy/MM/dd HH:mm:ss)	Yes

Response (json)：

Fields		Type	Description
ReturnCode		int	Return Code
Result		List<Data>	Data list
|--	Sn	string	S/N
|--	UploadTime	datetime	Data upload time
|--	Ppv1	decimal	PV1 power
|--	Ppv2	decimal	PV2 power
|--	Upv1	decimal	PV1 voltage
|--	Upv2	decimal	PV2 voltage
|--	Ua	decimal	Grid voltage 1
|--	Ub	decimal	Grid voltage 2
|--	Uc	decimal	Grid voltage 3
|--	Fac	decimal	Grid Frequency
|--	Ubus	decimal	Busbar voltage
|--	PrealL1	decimal	L1 power of inverter
|--	PrealL2	decimal	L2 power of inverter
|--	PrealL3	decimal	L2 power of inverter
|--	Tinv	decimal	Inverter core temperature
|--	PacL1	decimal	EMS control power commands 1
|--	PacL2	decimal	EMS control power commands 2
|--	PacL3	decimal	EMS control power commands 3
|--	InvWorkMode	int	Grid state: 1：wait；2：online；4：battery mode；8：bypass；16：Fault mode；32：VF start mode
|--	EpvTotal	decimal	PV generation energy
|--	Einput	decimal	Grid consumption energy
|--	Eoutput	decimal	Feed-in energy
|--	Echarge	decimal	Battery charging energy
|--	PmeterL1	decimal	AC meter power L1
|--	PmeterL2	decimal	AC meter power L2

|--	PmeterL3	decimal	AC meter power L3
|--	PmeterDC	decimal	DC meter power
|--	Pbat	decimal	Batter power
|--	SOC	decimal	State of charge
|--	BatV	decimal	Battery voltage
|--	BatC	decimal	Battery current
|--	FlagBms	char	BMS Flag
|--	BmsWork	int	BMS State: 0:standby 1:work 2:parallel 4: maintenance
|--	Pcharge	int	BMS allowed discharging power
|--	Pdischarge	int	BMS allowed discharging power
|--	BmsRelay	nvarchar	Relay status: XXX： Represents the state of the three relays
|--	BmsNum	nvarchar	Number of battery modules
|--	VcellLow	nvarchar	battery modules ID + battery identifier+ Minimum voltage
|--	VcellHigh	nvarchar	battery modules ID + battery identifier +Maximum voltage
|--	TcellLow	nvarchar	battery modules ID + Sampling point number + Minimum temperature
|--	TcellHigh	nvarchar	battery modules ID D+ Sampling point number + Maximum temperature
|--	IdTempLover	nvarchar	Low temperature Warning battery number
|--	IdTempEover	nvarchar	High temperature Warning battery number
|--	IdTempediffe	nvarchar	Temperature difference Warning battery number
|--	IdChargcurre	nvarchar	Charging current Warning battery number
|--	IdDischcurre	nvarchar	Discharging current Warning battery number
|--	IdCellvolover	nvarchar	Single cell overvoltage Warning battery number
|--	IdCellvollower	nvarchar	Single cell undervoltage Warning battery number
|--	IdSoclower	nvarchar	SOC critical low warning battery number
|--	IdCellvoldiffe	nvarchar	Single cell voltage difference Warning battery number
|--	BatC1	decimal	Battery 1 Current
|--	BatC2	decimal	Battery 2 Current
|--	BatC3	decimal	Battery 3 Current
|--	BatC4	decimal	Battery 4 Current
|--	BatC5	decimal	Battery 5 Current
|--	BatC6	decimal	Battery 6 Current
|--	ErrInv	nvarchar	Inverter fault
|--	WarInv	nvarchar	Inverter warning
|--	ErrEms	nvarchar	EMS fault
|--	ErrBms	nvarchar	Battery fault
|--	ErrMeter	nvarchar	Meter fault
|--	ErrBackupBox	nvarchar	BackupBox fault
|--	EGridCharge	nvarchar	Gird charged energy
|--	EmsStatus	nvarchar	Ems status
|--	EDischarge	decimal	Battery discharged energy
|--	SOC1	decimal	Battery 1 SOC

|--	SOC2	decimal	Battery 2 SOC
|--	SOC3	decimal	Battery 3 SOC
|--	SOC4	decimal	Battery 4 SOC
|--	SOC5	decimal	Battery 5 SOC
|--	SOC6	decimal	Battery 6 SOC
|--	VcellLowValue	decimal	Minimum voltage of single cell
|--	VcellHighValue	decimal	Maximum voltage of single cell
|--	TcellLowValue	decimal	Minimum temperature of single cell
|--	TcellHighValue	decimal	Maximum temperature of single cell
|--	InvBatV	decimal	Inverter sampling battery voltage
|--	BmsShutdown	int	Battery shutdown fault code
|--	BmuRelay	int	Relay status of SOC
|--	BmsHardVer1	int	Hardware version of the first set of battery cabinets（4 digits for a battery, 6 batteries in total）
|--	BmsHardVer2	int	Hardware version of the first set of battery cabinets（4 digits for a battery, 6 batteries in total）
|--	BmsHardVer3	int	Hardware version of the third set of battery cabinets（4 digits for a battery, 6 batteries in total）
|--	DispatchSwitch	decimal	Dispatching enable switch
|--	Pdispatch	decimal	Dispatching power
|--	DispatchSoc	decimal	SOC of dispatching
|--	DispatchMode	decimal	Dispatching Mode

6、Acquire data of system power

Website: https://api.alphaess.com/ras/v2/GetLastPowerData

Remark: This access is used to acquire data of system power. (one data per 10 seconds)

Http request method: POST https://api.alphaess.com/ras/v2/GetLastPowerData

Request parameter:

Parameter	Type	Description	Requires
api_account	string	Api account	Yes
timestamp	long	Unix timestamp is used to confirm validity of your request. If the timespan between request timestamp and server timestamp exceeds 300 seconds, then the request will be rejected. How to calculate the unix timestamp?	Yes

		(DateTime.UtcNow-new DateTime(1970,1,1)).TotalSeconds	
sign	string	Rule of signing: format request data (except sign) and secret key(provided by Alpha) using Key=Value, splice the keys arranged by sort ascending into a word string, and then use MD5 digest for passwords with 32-bit hex encoding. Please refer to the Signature Algorithm	Yes
Token	string	Authentication token	Yes
Sn	string	If S/N is entered then date of the device relating to this S/N will be provided otherwise all date of the devices of the particular user will be provided.	No

Response (json)：

Fields		Type	Description
ReturnCode		int	Return Code
Result		List<Data>	Data list
|--	Sn	string	S/N
|--	UploadTime	datetime	Data upload time
|--	Ppv1	decimal	PV1 power
|--	Ppv2	decimal	PV2 power
|--	PrealL1	decimal	L1 power of inverter
|--	PrealL2	decimal	L2 power of inverter
|--	PrealL3	decimal	L3 power of inverter
|--	PmeterL1	decimal	AC meter power L1
|--	PmeterL2	decimal	AC meter power L2
|--	PmeterL3	decimal	AC meter power L3
|--	PmeterDC	decimal	DC meter power
|--	Pbat	decimal	Batter power
|--	Sva	decimal	Apparent power of inverter

7、 Send system dispatch commands

Website: https://api.alphaess.com/ras/v2/RemoteDispatch

Remark: This access is used to send system dispatch commands. (The command will automatically stop after 90 seconds)

Http request method: POST https://api.alphaess.com/ras/v2/RemoteDispatch

Request parameter:

Parameter	Type	Description	Requires
api_account	string	Api account	Yes
timestamp	long	Unix timestamp is used to confirm validity of your request. If the timespan between request timestamp and server timestamp exceeds 300 seconds, then the request will be rejected.	Yes

		How to calculate the unix timestamp? (DateTime.UtcNow-new DateTime(1970,1,1)).TotalSeconds	
sign	string	Rule of signing: format request data (except sign) and secret key(provided by Alpha) using Key=Value, splice the keys arranged by sort ascending into a word string, and then use MD5 digest for passwords with 32-bit hex encoding. Please refer to the Signature Algorithm	Yes
Token	string	Authentication token	Yes
Sn	string	If S/N is entered then date of the device relating to this S/N will be provided otherwise all date of the devices of the particular user will be provided.	No
ActivePower	int	Inverter storage active power Remark : 1w/bit offset：32000 charge:<32000 discha:>32000 example： send Pset =34500, corresponding to the discharge power of 2500w	Yes
ReactivePower	int	Inverter storage reactive power Remark: 1w/bit offset:32000 charge:<32000 discha:>32000 example： send Pset =34500, corresponding to the discharge power of 2500w	Yes
SOC	decimal	Force storage to reach SOC Remark : 0.4%/bit offset：0 example：Send SOC =95, corresponding to the SOC of 38%.	Yes
Status	Int	power grid dispatching(1:start , 0:stop)	Yes
ControlMode	Int	1:Battery only charges from PV 2:State of Charge control 3:Load Following	Yes

Response (json)：

Fields		Type	Description
ReturnCode		int	Return Code

8、ReturnCode

Return Code	Description
0	Data successfully downloaded
1	Error
2	Required fields not filled
3	Invalid timestamp
4	Authentication unsuccessful
5	User name already exists
6	License already exists
7	SN already exists
8	Invalid API account
9	Invalid User
10	E-mail address and username do not match our records.

11	Your username or password is incorrect.
12	Invalid SN
13	The command Dispatch failed.
14	Login timeout
15	You don't have appropriate permission to perform this operation.
16	Abnormal login
-1	Unknown mistake

9、Encryption Algorithm

```
public static string EncryptStringToBytes_Aes(string plainText)
{
    string encrypted = string.Empty;
    byte[] clearBytes = Encoding. UTF8.GetBytes(plainText);
    using (Aes aesAlg = Aes.Create())
    {
        byte[] k;
        byte[] iv;
            byte[] bytes = Encoding.UTF8.GetBytes(secretKey);
        k = SHA256.Create().ComputeHash(bytes);
        iv = MD5.Create().ComputeHash(bytes);
        aesAlg.Key = k;
        aesAlg.IV = iv;

        ICryptoTransform encryptor = aesAlg.CreateEncryptor(aesAlg.Key, aesAlg.IV);

        using (MemoryStream msEncrypt = new MemoryStream())
        {
                using (CryptoStream csEncrypt = new CryptoStream(msEncrypt, encryptor, CryptoStreamMode.Write))
                {
                        csEncrypt.Write(clearBytes, 0, clearBytes.Length);
                }
                encrypted = Convert.ToBase64String(msEncrypt.ToArray());
        }
    }

    return encrypted;
}
```

10、Signature Algorithm

(this is to show how users retrieve password, others are similar)

```
SortedList slstParams = new SortedList();
slstParams.Add("api_account ", api_account.ToString());
slstParams.Add("timestamp", timestamp.ToString());
slstParams.Add("username", username);
slstParams.Add("email", email);
slstParams.Add("secretkey", secret);

StringBuilder strParams = new StringBuilder();

for (var i = 0; i < slstParams.Count; i++)
{
        strParams.AppendFormat("{0}={1}", slstParams.GetKey(i), slstParams.GetByIndex(i));
}

var md5Hash = MD5Helper.GenerateMD5Hash(strParams.ToString());
```

Note: if the parameter is empty, submit empty string ("") if it's string and 0 if it's number.

11、MD5Helper

```
public class MD5Helper
{
        public static string GenerateMD5Hash(string input)
        {
                if (string.IsNullOrEmpty(input))
                {
                        throw new ArgumentException("argument cannot be null", "input");
                }

                using (MD5 md5Hash = MD5.Create())
                {
                        byte[] data = md5Hash.ComputeHash(Encoding.UTF8.GetBytes(input));
                        StringBuilder sb = new StringBuilder();
```

```
for (int i = 0; i < data.Length; i++)
{
        sb.Append(data[i].ToString("x2"));
}
return sb.ToString();
}
}
}
```