

Table of Contents

1 Account	1
2 Account Management API	2
2.1 Contents	2
2.2 Account Management	2
2.3 Account Attributes Management	2
2.4 Contacts Management	2
2.5 Users Management	3
2.6 Products Management	3
2.7 Products Bandwidth Management	3
2.8 Products Usage Management	3
2.9 Payment info	3
2.10 Transactions	3
2.11 Invoices	3
2.12 Invoice Transactions	3
2.13 Account Discount	3
2.14 Security/Firewall Group	3
2.15 See also	4
3 Accounts-accountId-action-interFranchiseMigration:GET	5
4 GET /accounts/{accountId}/action/interFranchiseMigration	6
4.1 Contents	6
4.2 Request	6
4.3 Response	6
4.4 Conditional Hypermedia	7
4.5 Examples	7
4.6 See also	8
5 Accounts-accountId-action:POST	9
6 POST /accounts/{accountId}/action	10
6.1 Contents	10
6.2 Request	10
6.3 Response	10
6.4 Examples	10
6.5 See also	11
7 Accounts-accountId-attributes-attribute:DELETE	12
8 DELETE /accounts/{accountId}/attributes/{attribute}	13
8.1 Contents	13
8.2 Request	13
8.3 Response	13
8.4 Examples	13
8.5 See also	14
9 Accounts-accountId-attributes-attribute:GET	15
10 GET /accounts/{accountId}/attributes/{attribute}	16
10.1 Contents	16
10.2 Request	16
10.3 Response	16
10.4 Examples	17
10.5 See also	17
11 Accounts-accountId-attributes-attribute:PUT	18
12 PUT /accounts/{accountId}/attributes/{attribute}	19
12.1 Contents	19
12.2 Request	19
12.3 Response	19
12.4 Examples	19
12.5 See also	20
13 Accounts-accountId-attributes:GET	21
14 GET /accounts/{accountId}/attributes	22
14.1 Contents	22
14.2 Request	22
14.3 Response	22
14.4 Examples	23
14.5 See also	23
15 Accounts-accountId-attributes:POST	24
16 POST /accounts/{accountId}/attributes	25
16.1 Contents	25
16.2 Request	25
16.3 Response	25
16.4 Examples	25
16.5 See also	26

Table of Contents

17 Accounts-accountId-Contacts-contactType:GET	27
18 GET /accounts/{accountId}/contacts/{contactType}	28
18.1 Contents	28
18.2 Request	28
18.3 Response	28
18.4 Examples	29
18.5 See also	30
19 Accounts-accountId-Contacts-contactType:PUT	31
20 PUT /accounts/{accountId}/contacts/{contactType}	32
20.1 Contents	32
20.2 Request	32
20.3 Response	33
20.4 Examples	33
20.5 See also	35
21 Accounts-accountId-Contacts:GET	36
22 GET /accounts/{accountId}/contacts	37
22.1 Contents	37
22.2 Request	37
22.3 Response	37
22.4 Examples	39
22.5 See also	40
23 Accounts-accountId-Contacts:PUT	41
24 PUT /accounts/{accountId}/contacts	42
24.1 Contents	42
24.2 Request	42
24.3 Response	43
24.4 Examples	44
24.5 See also	45
25 Accounts-accountId-discount:GET	46
26 GET /accounts/{accountId}/discount	47
26.1 Contents	47
26.2 Request	47
26.3 Response	47
26.4 Examples	47
26.5 See also	48
27 Accounts-accountId-invoices-invoiceId-transactions:GET	49
28 GET /accounts/{accountId}/invoices/{invoiceId}/transactions	50
28.1 Contents	50
28.2 Request	50
28.3 Response	50
28.4 Examples	51
28.5 See also	51
29 Accounts-accountId-invoicesInfo-filtered-status:GET	52
30 GET /accounts/{accountId}/invoice?filterStatus={filterStatus}	53
30.1 Contents	53
30.2 Request	53
30.3 Response	53
30.4 Examples	54
30.5 See also	54
31 Accounts-accountId-invoicesInfo-filtered-type:GET	55
32 GET /accounts/{accountId}/invoice?filterStatus={filterStatus}	56
32.1 Contents	56
32.2 Request	56
32.3 Response	56
32.4 Examples	57
32.5 See also	57
33 Accounts-accountId-invoicesInfo-invoiceId:GET	58
34 GET /accounts/{accountId}/invoces/{invoiceId}	59
34.1 Contents	59
34.2 Request	59
34.3 Response	59
34.4 Examples	60
34.5 See also	60
35 Accounts-accountId-invoicesInfo-sorted:GET	61

Table of Contents

36 GET /accounts/{accountId}/invoice?sortField={sortField}	62
36.1 Contents.....	62
36.2 Request.....	62
36.3 Response.....	62
36.4 Examples.....	63
36.5 See also.....	63
37 Accounts-accountId-invoicesInfo:GET	64
38 GET /accounts/{accountId}/invoices	65
38.1 Contents.....	65
38.2 Request.....	65
38.3 Response.....	65
38.4 Examples.....	66
38.5 See also.....	66
39 Accounts-accountId-paymentInfo-default:DELETE	67
40 DELETE /accounts/{accountId}/paymentInfo/default	68
40.1 Contents.....	68
40.2 Request.....	68
40.3 Response.....	68
40.4 See also.....	68
41 Accounts-accountId-paymentInfo-default:PUT	69
42 PUT /accounts/{accountId}/paymentInfo/default	70
42.1 Contents.....	70
42.2 Request.....	70
42.3 Response.....	70
42.4 See also.....	70
43 Accounts-accountId-paymentInfo-paymentInfoId-payment:POST	71
44 POST /accounts/{accountId}/paymentInfo/{paymentInfoId}/payment	72
44.1 Contents.....	72
44.2 Request.....	72
44.3 Response.....	72
44.4 Examples.....	73
44.5 See also.....	73
45 Accounts-accountId-paymentInfo-paymentInfoId:DELETE	74
46 DELETE /accounts/{accountId}/paymentInfo/{paymentInfoId}	75
46.1 Contents.....	75
46.2 Request.....	75
46.3 Response.....	75
46.4 See also.....	75
47 Accounts-accountId-paymentInfo-paymentInfoId:GET	76
48 GET /accounts/{accountId}/paymentInfo/{paymentInfoId}	77
48.1 Contents.....	77
48.2 Request.....	77
48.3 Response.....	77
48.4 See also.....	78
49 Accounts-accountId-paymentInfo:GET	79
50 GET /accounts/{accountId}/paymentInfo	80
50.1 Contents.....	80
50.2 Request.....	80
50.3 Response.....	80
50.4 See also.....	81
51 Accounts-accountId-paymentInfo:POST	82
52 POST /accounts/{accountId}/paymentInfo	83
52.1 Contents.....	83
52.2 Request.....	83
52.3 Response.....	84
52.4 Examples.....	85
52.5 See also.....	85
53 Accounts-accountId-products-productId-action:POST	86
54 POST /accounts/{accountId}/products/{productId}/action	87
54.1 Contents.....	87
54.2 Request.....	87
54.3 Response.....	87
54.4 See also.....	87
55 Accounts-accountId-products-productId-bandwidth-daily:GET	89

Table of Contents

56 GET /accounts/{accountId}/products/{productId}/bandwidth/daily?filterType={filterType}&from={filterStartDate}&to={filterEndDate}.....	90
56.1 Contents.....	90
56.2 Request.....	90
56.3 Response.....	90
56.4 Examples.....	91
56.5 See also.....	91
57 Accounts-accountId-products-productId-bandwidth-monthly:GET.....	92
58 GET /accounts/{accountId}/products/{productId}/bandwidth/monthly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}..	93
58.1 Contents.....	93
58.2 Request.....	93
58.3 Response.....	93
58.4 Examples.....	94
58.5 See also.....	95
59 Accounts-accountId-products-productId-bandwidth-weekly:GET.....	96
60 GET /accounts/{accountId}/products/{productId}/bandwidth/weekly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}....	97
60.1 Contents.....	97
60.2 Request.....	97
60.3 Response.....	97
60.4 Examples.....	98
60.5 See also.....	99
61 Accounts-accountId-products-productId-bandwidth-yearly:GET.....	100
62 GET /accounts/{accountId}/products/{productId}/bandwidth/yearly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}....	101
62.1 Contents.....	101
62.2 Request.....	101
62.3 Response.....	101
62.4 Examples.....	102
62.5 See also.....	102
63 Accounts-accountId-products-productId-services-serviceId-action:GET.....	103
64 POST /accounts/{accountId}/products/{productId}/services/{serviceId}/action.....	104
64.1 Contents.....	104
64.2 Request.....	104
64.3 Response.....	104
64.4 See also.....	105
65 Accounts-accountId-products-productId-services-serviceId-action:POST.....	106
66 POST /accounts/{accountId}/products/{productId}/services/{serviceId}/action.....	107
66.1 Contents.....	107
66.2 Request.....	107
66.3 Response.....	107
66.4 See also.....	108
67 Accounts-accountId-products-productId-services-serviceId-getTargetServiceSpecifications:GET.....	109
68 GET /accounts/{accountId}/products/{productId}/services/{serviceId}?getTargetServiceSpecifications=1.....	110
68.1 Contents.....	110
68.2 Request.....	110
68.3 Response.....	110
68.4 See also.....	111
69 Accounts-accountId-products-productId-services-serviceId:GET.....	112
70 GET /accounts/{accountId}/products/{productId}/services/{serviceId}.....	113
70.1 Contents.....	113
70.2 Request.....	113
70.3 Response.....	113
70.4 See also.....	114
71 Accounts-accountId-products-productId-services-serviceId:PUT.....	115
72 PUT /accounts/{accountId}/products/{productId}/services/{serviceId}'.....	116
72.1 Contents.....	116
72.2 Request.....	116
72.3 Response.....	116
72.4 See also.....	116
73 Accounts-accountId-products-productId-services:GET.....	117
74 GET /accounts/{accountId}/products/{productId}/services.....	118
74.1 Contents.....	118
74.2 Request.....	118
74.3 Response.....	118
74.4 See also.....	119
75 Accounts-accountId-products-productId-usage:GET.....	120

Table of Contents

76 GET /accounts/{accountId}/products/{productId}/usage?from={fromDate}&to={toDate}&rate=true	121
76.1 Contents.....	121
76.2 Request.....	121
76.3 Response.....	121
76.4 Examples.....	123
76.5 See also.....	124
77 Accounts-accountId-products-productId:GET	125
78 GET /accounts/{accountId}/products/{productId}	126
78.1 Contents.....	126
78.2 Request.....	126
78.3 Response.....	126
78.4 See also.....	127
79 Accounts-accountId-products-productId:PUT	128
80 PUT /accounts/{accountId}/products/{productId}'	129
80.1 Contents.....	129
80.2 Request.....	129
80.3 Response.....	129
80.4 See also.....	129
81 Accounts-accountId-products:GET	130
82 GET /accounts/{accountId}/products	131
82.1 Contents.....	131
82.2 Request.....	131
82.3 Response.....	131
82.4 Alternative with partner Id.....	132
82.5 See also.....	132
83 Accounts-accountId-transactionsInfo-filtered-type:GET	133
84 GET /accounts/{accountId}/transactions?filterType={filterType}	134
84.1 Contents.....	134
84.2 Request.....	134
84.3 Response.....	134
84.4 Examples.....	135
84.5 See also.....	135
85 Accounts-accountId-transactionsInfo-sorted:GET	136
86 GET /accounts/{accountId}/transactions?sortField={sortField}&sortOrder={sortOrder}	137
86.1 Contents.....	137
86.2 Request.....	137
86.3 Response.....	137
86.4 Examples.....	138
86.5 See also.....	138
87 Accounts-accountId-transactionsInfo:GET	139
88 GET /accounts/{accountId}/transactions	140
88.1 Contents.....	140
88.2 Request.....	140
88.3 Response.....	140
88.4 Examples.....	141
88.5 See also.....	141
89 Accounts-accountId-user:PUT	142
90 PUT /accounts/{accountId}/user	143
90.1 Contents.....	143
90.2 Request.....	143
90.3 Response.....	143
90.4 Examples.....	144
90.5 See also.....	145
91 Accounts-accountId-users-userName-roles:GET	146
92 GET /accounts/{accountId}/users/{userName}/roles	147
92.1 Contents.....	147
92.2 Request.....	147
92.3 Response.....	147
92.4 See also.....	147
93 Accounts-accountId-users-userName-roles:PUT	148
94 PUT /accounts/{accountId}/users/{userName}/roles	149
94.1 Contents.....	149
94.2 Request.....	149
94.3 Response.....	149
94.4 See also.....	149

Table of Contents

95 Accounts-accountId-users-userName:DELETE	150
96 DELETE /accounts/{accountId}/users/{userName}	151
96.1 Contents.....	151
96.2 Request.....	151
96.3 Response.....	151
96.4 See also.....	151
97 Accounts-accountId-users-userName:GET	152
98 GET /accounts/{accountId}/users/{userName}	153
98.1 Contents.....	153
98.2 Request.....	153
98.3 Response.....	153
98.4 See also.....	153
99 Accounts-accountId-users-userName:PUT	155
100 PUT /accounts/{accountId}/users/{userName}	156
100.1 Contents.....	156
100.2 Request.....	156
100.3 Response.....	156
100.4 See also.....	157
101 Accounts-accountId-users:GET	158
102 GET /accounts/{accountId}/users	159
102.1 Contents.....	159
102.2 Request.....	159
102.3 Response.....	159
102.4 See Also.....	160
103 Accounts-accountId-users:POST	161
104 POST /accounts/{accountId}/users	162
104.1 Contents.....	162
104.2 Request.....	162
104.3 Response.....	162
104.4 Allowed Security Questions.....	163
104.5 See also.....	163
105 Accounts-accountId:DELETE	164
106 DELETE /accounts/{accountId}	165
106.1 Contents.....	165
106.2 Request.....	165
106.3 Response.....	165
106.4 Alternative with partner Id.....	165
106.5 See also.....	165
107 Accounts-accountId:GET	166
108 GET /accounts/{accountId}	167
108.1 Contents.....	167
108.2 Request.....	167
108.3 Response.....	167
108.4 Conditional Hypermedia.....	168
108.5 Alternative with partner Id.....	168
108.6 See also.....	169
109 Accounts-accountId:PUT	170
110 PUT /accounts/{accountId}	171
110.1 Contents.....	171
110.2 Request.....	171
110.3 Response.....	171
110.4 Alternative with partner Id.....	171
110.5 See also.....	172
111 Accounts:GET	173
112 GET /accounts	174
112.1 Contents.....	174
112.2 Request.....	174
112.3 Response.....	174
112.4 Examples.....	175
112.5 See also.....	176
113 Avas-SSO:GET	177
114 GET /accounts/{accountId}/avas/{domain}/SSOLink	178
114.1 Contents.....	178
114.2 Request.....	178
114.3 Response.....	178
114.4 Examples.....	178

Table of Contents

115 Email Validation:POST	180
115.1 Contents.....	180
115.2 Request.....	180
115.3 Response.....	180
115.4 Example.....	180
115.5 See also.....	181
116 Initiate-Email-Verification-Action:POST	182
117 POST /accounts/{account_number}/contacts/action	183
117.1 Contents.....	183
117.2 Request.....	183
117.3 Response.....	183
117.4 Examples.....	183
117.5 See also.....	184
118 PartnerAccounts-partnerAccountId-attributes-attribute:DELETE	185
119 DELETE /partnerAccounts/{partnerAccountId}/attributes/{attribute}	186
119.1 Contents.....	186
119.2 Request.....	186
119.3 Response.....	186
119.4 Examples.....	186
119.5 See also.....	187
120 PartnerAccounts-partnerAccountId-attributes-attribute:GET	188
121 GET /partnerAccounts/{partnerAccountId}/attributes/{attribute}	189
121.1 Contents.....	189
121.2 Request.....	189
121.3 Response.....	189
121.4 Examples.....	190
121.5 See also.....	190
122 PartnerAccounts-partnerAccountId-attributes-attribute:PUT	191
123 PUT /partnerAccounts/{partnerAccountId}/attributes/{attribute}	192
123.1 Contents.....	192
123.2 Request.....	192
123.3 Response.....	192
123.4 Examples.....	192
123.5 See also.....	193
124 PartnerAccounts-partnerAccountId-attributes:GET	194
125 GET /partnerAccounts/{partnerAccountId}/attributes	195
125.1 Contents.....	195
125.2 Request.....	195
125.3 Response.....	195
125.4 Examples.....	196
125.5 See also.....	196
126 PartnerAccounts-partnerAccountId-attributes:POST	197
127 POST /partnerAccounts/{partnerAccountId}/attributes	198
127.1 Contents.....	198
127.2 Request.....	198
127.3 Response.....	198
127.4 Examples.....	199
127.5 See also.....	199
128 ServiceAttributes-core.cloud.hsc	200
129 Cloud Compute Service Attributes	201
129.1 Contents.....	201
129.2 Attributes Body.....	201
129.3 See also.....	201
130 ServiceAttributes-core.dedserver	202
131 Dedicated Server Service Attributes	203
131.1 Contents.....	203
131.2 Attributes Body.....	203
131.3 See also.....	203
132 ServiceAttributes-core.dnr	204
133 Domain Name Registration Service Attributes	205
133.1 Contents.....	205
133.2 Attributes Body.....	205
133.3 See also.....	206
134 ServiceAttributes-core.fax.number	207

Table of Contents

135 Fax Over Email Number Attributes.....	208
135.1 Contents.....	208
135.2 Attributes Body.....	208
135.3 See also.....	208
136 ServiceAttributes-core.fax.user.....	209
137 Fax Over Email User Attributes.....	210
137.1 Contents.....	210
137.2 Attributes Body.....	210
137.3 See also.....	210
138 ServiceAttributes-core.online.backup.account.....	211
139 Online Backup Account Attributes.....	212
139.1 Contents.....	212
139.2 Attributes Body.....	212
139.3 See also.....	212
140 ServiceAttributes-core.sharepoint.....	213
141 Sharepoint Service Attributes.....	214
141.1 Contents.....	214
141.2 Attributes Body.....	214
141.3 See also.....	214
142 ServiceAttributes-core.sharepoint2007.....	215
143 SharePoint 2007 Attributes.....	216
143.1 Contents.....	216
143.2 Attributes Body.....	216
143.3 See also.....	216
144 ServiceAttributes-core.spamexperts.....	217
145 AVAS Service Attributes.....	218
145.1 Contents.....	218
145.2 Attributes Body.....	218
145.3 See also.....	218
146 ServiceAttributes-core.subdomain.....	219
147 Sub-Domain Registration Service Attributes.....	220
147.1 Contents.....	220
147.2 Attributes Body.....	220
147.3 See also.....	220
148 ServiceAttributes-core.vps.virtuozzo.server.....	221
149 VPS Virtuozzo Server Attributes.....	222
149.1 Contents.....	222
149.2 Attributes Body.....	222
149.3 See also.....	222
150 ServiceAttributes-core.webhosting.....	223
151 Web Hosting Service Attributes.....	224
151.1 Contents.....	224
151.2 Attributes Body.....	224
151.3 See also.....	224
152 ServiceAttributes-core.websplanet.....	225
153 Websplanet Attributes.....	226
153.1 Contents.....	226
153.2 Attributes Body.....	226
153.3 See also.....	226
154 ServiceAttributes-dedicated.server.firewall.....	227
155 Dedicated Server Firewall Service Attributes.....	228
155.1 Contents.....	228
155.2 Attributes Body.....	228
155.3 See also.....	228
156 ServiceAttributes-dnr.feature.domain.guard.....	229
157 DomainGuard Service Attributes.....	230
157.1 Contents.....	230
157.2 Attributes Body.....	230
157.3 See also.....	230
158 ServiceAttributes-dnr.feature.private.registration.....	231

Table of Contents

159 WhoisProtector(Private Registration) Service Attributes.....	232
159.1 Contents.....	232
159.2 Attributes Body.....	232
159.3 See also.....	232
160 ServiceAttributes-mcafee.....	233
161 Excel Micro Service Attributes.....	234
161.1 See also.....	234
162 ServiceAttributesStructure.....	235
162.1 See also.....	235
163 ZenDesk-SSO-Link-Per-Contact-Type:GET.....	236
164 GET /accounts/{accountId}/contacts/{contactType}/supportSSOLink.....	237
164.1 Contents.....	237
164.2 Request.....	237
164.3 Response.....	237
164.4 Examples.....	237
164.5 See also.....	238

1 Account

2 Account Management API

The account management API provides methods for managing the account contacts, users and products. Clients can update the status of the account products as well as change (upgrade/downgrade) them.

2.1 Contents

- 1 Account Management API
 - ◆ 1.1 Account Management
 - ◆ 1.2 Account Attributes Management
 - ◆ 1.3 Contacts Management
 - ◇ 1.3.1 Actions
 - ◇ 1.3.2 ZenDesk SSO Link
 - ◆ 1.4 Users Management
 - ◆ 1.5 Products Management
 - ◆ 1.6 Products Bandwidth Management
 - ◆ 1.7 Products Usage Management
 - ◆ 1.8 Payment info
 - ◆ 1.9 Transactions
 - ◆ 1.10 Invoices
 - ◆ 1.11 Invoice Transactions
 - ◆ 1.12 Account Discount
 - ◆ 1.13 Security/Firewall Group
 - ◆ 1.14 See also

[download this selection of articles as a PDF book](#)

2.2 Account Management

- GET /accounts - Returns list of accounts authorized for access with the provided credentials
- GET /accounts/{accountId} - Returns information about an account
- GET /partnerAccounts/{partnerAccountId} - Returns information about an account specified by its partner Id
- PUT /accounts/{accountId} - Updates account status
- PUT /partnerAccounts/{partnerAccountId} - Updates account status for account specified by its partner Id
- DELETE /accounts/{accountId} - Purges an account from the system
- DELETE /partnerAccounts/{partnerAccountId} - Purges an account specified by its partner Id from the system
- GET /accounts/{accountId}/action/interFranchiseMigration - Returns information about an account's migration to another franchise
- POST /accounts/{accountId}/action - Initiate an action on an account

2.3 Account Attributes Management

- GET /accounts/{accountId}/attributes - Returns list of attributes
- GET /partnerAccounts/{partnerAccountId}/attributes - Returns list of attributes using Partner Account Id
- POST /accounts/{accountId}/attributes - Creates new attribute
- POST /partnerAccounts/{partnerAccountId}/attributes - Creates new attribute using Partner Account Id
- GET /accounts/{accountId}/attributes/{attribute} - Gets attribute using Account ID
- GET /partnerAccounts/{partnerAccountId}/attributes/{attribute} - Gets attribute using Partner Account Id
- PUT /accounts/{accountId}/attributes/{attribute} - Updates attribute value
- PUT /partnerAccounts/{partnerAccountId}/attributes/{attribute} - Updates attribute value using Partner Account Id
- DELETE /accounts/{accountId}/attributes/{attribute} - Deletes attribute
- DELETE /partnerAccounts/{partnerAccountId}/attributes/{attribute} - Deletes attribute using Partner Account Id

2.4 Contacts Management

- GET /accounts/{accountId}/contacts - Returns all contacts for an account
- GET /accounts/{accountId}/contacts/{contactType} - Returns specific contact type for an account
- PUT /accounts/{accountId}/contacts - Set all contacts for an account
- PUT /accounts/{accountId}/contacts/{contactType} - Set specific contact type for an account

2.4.1 Actions

- POST /accounts/{account_number}/contacts/action - Initiate Email Verification Action

2.4.2 ZenDesk SSO Link

- GET /accounts/{accountId}/contacts/{contactType}/supportSSOLink - Returns Zendesk SSO Link Per Contact Type

2.5 Users Management

- GET /accounts/{accountId}/users - Returns a list of users in the account
- POST /accounts/{accountId}/users - Creates a new user in the account
- GET /accounts/{accountId}/user/{userName} - Returns the details of a specific user in the account
- PUT /accounts/{accountId}/user/{userName} - Updates the user details (password, security question/answer)
- PUT /accounts/{accountId}/user - (Deprecated) Updates the user details (password, security question/answer) of a user
- DELETE /accounts/{accountId}/user/{userName} - Deletes a user
- GET /accounts/{accountId}/user/{userName}/roles - Returns the list of roles of a user
- PUT /accounts/{accountId}/user/{userName}/roles - Updates the list of roles of a user

2.6 Products Management

- GET /accounts/{accountId}/products - Gets all products for an account
- GET /accounts/{accountId}/products/{productId} - Get information about a product
- PUT /accounts/{accountId}/products/{productId} - Update the status of a product
- POST /accounts/{accountId}/products/{productId}/action - Perform an action (e.g: upgrade/downgrade) for a product
- GET /accounts/{accountId}/products/{productId}/services - List services for a specific product
- GET /accounts/{accountId}/products/{productId}/services/{serviceld} - Get information about a specific service
- PUT /accounts/{accountId}/products/{productId}/services/{serviceld} - Update the status of a specific service
- GET /accounts/{accountId}/products/{productId}/services/{serviceld}/?getTargetServiceSpecifications=1 - Get allowed specific service changes
- POST /accounts/{accountId}/products/{productId}/services/{serviceld}/action - Perform an action (e.g: upgrade/downgrade) for a specific service

2.7 Products Bandwidth Management

- GET /accounts/{accountId}/products/{productId}/bandwidth/daily?filterType={filterType}&from={filterStartDate}&to={filterEndDate} - Get a report on the daily bandwidth usage of the product
- GET /accounts/{accountId}/products/{productId}/bandwidth/weekly?filterType={filterType}&from={filterStartDate}&to={filterEndDate} - Get a report on the weekly bandwidth usage of the product
- GET /accounts/{accountId}/products/{productId}/bandwidth/monthly?filterType={filterType}&from={filterStartDate}&to={filterEndDate} - Get a report on the monthly bandwidth usage of the product
- GET /accounts/{accountId}/products/{productId}/bandwidth/yearly?filterType={filterType}&from={filterStartDate}&to={filterEndDate} - Get a report on the yearly bandwidth usage of the product

Each bandwidth usage is filtered by the traffic type(www or ftp), the start date and the end date in ISO format(eg. 2013-20-02).

2.8 Products Usage Management

- GET /accounts/{accountId}/products/{productId}/usage - Get an aggregated services usage report of a product

2.9 Payment info

- GET /accounts/{accountId}/paymentInfo - Get a list of active payment methods
- POST /accounts/{accountId}/paymentInfo - Add a payment method for a given account
- PUT /accounts/{accountId}/paymentInfo/default - Set the default payment method
- DELETE /accounts/{accountId}/paymentInfo/default - Unset the default payment method
- GET /accounts/{accountId}/paymentInfo/{paymentInfoId} - Get information about an active payment method
- DELETE /accounts/{accountId}/paymentInfo/{paymentInfoId} - Remove an active payment method
- POST /accounts/{accountId}/paymentInfo/{paymentInfoId}/payment - Make a one-time payment toward the account balance

2.10 Transactions

- GET /accounts/{accountId}/transaction - Get a list of account's transactions
- GET /accounts/{accountId}/transaction?filterType={filterType} - Get a list of account's transactions filtered by Transaction type
- GET /accounts/{accountId}/transaction?sortField={sortField} - Get a list of account's transactions sorted by sortField param

2.11 Invoices

- GET /accounts/{accountId}/invoices - Get a list of account's invoices
- GET /accounts/{accountId}/invoices?filterStatus={filterStatus} - Get a list of account's invoices filtered by Invoice status
- GET /accounts/{accountId}/invoices?sortField={sortField} - Get a list of account's invoices sorted by sortField param
- GET /accounts/{accountId}/invoices/{invoiceId} - Get invoice info or pdf

2.12 Invoice Transactions

- GET /accounts/{accountId}/invoices/{invoiceId}/transactions - Get a list of invoice transactions

2.13 Account Discount

- GET /accounts/{accountId}/discount - Get account level discount

2.14 Security/Firewall Group

A Security/Firewall Group defines firewall rules for your Server. You have the ability to configure your firewall connections and secure your network as required. You may have one or multiple servers behind a firewall group. Hostway provides a default firewall group, and typical firewall rules are already configured. You can create multiple firewall groups as needed.

- GET /accounts/{accountId}/os-security-groups - Get a list of account's firewall groups
- POST /accounts/{accountId}/os-security-groups - Create firewall group for account
- GET /accounts/{accountId}/os-security-groups/{groupId} - Get firewall group details
- PUT /accounts/{accountId}/os-security-groups/{groupId} - Update firewall group
- DELETE /accounts/{accountId}/os-security-groups/{groupId} - Delete firewall group
- POST /accounts/{accountId}/os-security-group-rules - Create firewall group rule

- [DELETE /accounts/{accountId}/os-security-group-rules/{ruleId}](#) - Delete firewall group rule

2.15 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

3 Accounts-accountId-action-interFranchiseMigration:GET

4 GET /accounts/{accountId}/action/interFranchiseMigration

Get details about an account's migration to another franchise

4.1 Contents

- 1 GET /accounts/{accountId}/action/interFranchiseMigration
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Conditional Hypermedia
 - ◆ 1.4 Examples
 - ◇ 1.4.1 Requesting the status of a migration that has been submitted but not started
 - ◇ 1.4.2 Requesting the status of a migration that is being processed (10% done)
 - ◇ 1.4.3 Requesting the status of a migration that is being processed (70% done)
 - ◇ 1.4.4 Requesting the status of a migration that is finished
 - ◇ 1.4.5 Requesting the status of a migration that was never started
 - ◇ 1.4.6 Requesting the status of a migration that does not exist
 - ◆ 1.5 See also

4.2 Request

GET /accounts/{accountId}/action/interFranchiseMigration

4.2.1 Parameters

accountId - *string*
The Id of the account.

4.2.2 URI Parameters

None.

4.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

4.3 Response

4.3.1 Status Code

200 OK
Success.

403 Forbidden
Forbidden access to accounts by a user not authorized for such access.

404 Not Found
The account Id was not found or a migration request has not been created for this account.

4.3.2 Response Body

```
{
  "status": "{status}",
  "progress": {progress},
  "links": [
    {"href": "https://api.hostway.com/accounts/{account-number}/action/interFranchiseMigration", "rel": "self"}
  ]
}
```

4.3.2.1 Parameters

status - *string*
The status of the migration. Valid values are **pending**, **complete**, **inprogress** or **unknown** - in case there is no InterFranchise migration started on the account, for this case the progress is None.

progress - *float*
A float from 0 to 1 representing the migration request's progress. The progress would indicated which is the current flow of the InterFranchise migration process - 0.1 would indicate that it is on the first flow, 0.9 on the last, 1 would indicated that the migration was completed. Can be *null* if the status is **unknown**.

newAccountId - *string*
The newly-generated accountId for the target account in the target franchise. Available only when the status is **complete**.

newFranchise - *string*
The target franchise this account was migrated to. Available only when the status is **complete**.

links - *list* [Hypermedia](#) for this resource. Links to self.

4.4 Conditional Hypermedia

Hypermedia fields marked as optional will appear on the basis of the accessing user's permissions.

4.5 Examples

4.5.1 Requesting the status of a migration that has been submitted but not started

Request

```
GET {{APIBaseURL}}/accounts/account123/action/interFranchiseMigration
```

Response

```
HTTP/1.1 200 OK
```

```
{
  "status": "pending",
  "progress": 0,
  "links": [
    { "href": "{{APIBaseURL}}/accounts/account123/action/interFranchiseMigration", "rel": "self" }
  ]
}
```

4.5.2 Requesting the status of a migration that is being processed (10% done)

Request

```
GET {{APIBaseURL}}/accounts/account123/action/interFranchiseMigration
```

Response

```
HTTP/1.1 200 OK
```

```
{
  "status": "inprogress",
  "progress": 0.1,
  "links": [
    { "href": "{{APIBaseURL}}/accounts/account123/action/interFranchiseMigration", "rel": "self" }
  ]
}
```

4.5.3 Requesting the status of a migration that is being processed (70% done)

Request

```
GET {{APIBaseURL}}/accounts/account123/action/interFranchiseMigration
```

Response

```
HTTP/1.1 200 OK
```

```
{
  "status": "inprogress",
  "progress": 0.7,
  "links": [
    { "href": "{{APIBaseURL}}/accounts/account123/action/interFranchiseMigration", "rel": "self" }
  ]
}
```

4.5.4 Requesting the status of a migration that is finished

Request

```
GET {{APIBaseURL}}/accounts/account123/action/interFranchiseMigration
```

Response

```
HTTP/1.1 200 OK
```

```
{
  "status": "complete",
  "progress": 1,
  "newAccountId": "acc853234524",
  "newFranchise": "domainpeople_us_main",
  "links": [
    { "href": "{{APIBaseURL}}/accounts/account123/action/interFranchiseMigration", "rel": "self" }
  ]
}
```

4.5.5 Requesting the status of a migration that was never started

Request

```
GET {{APIBaseURL}}/accounts/account123/action/interFranchiseMigration
```

Response

HTTP/1.1 200 OK

```
{
  "status": "unknown",
  "progress": null,
  "links": [
    { "href": "{{APIBaseURL}}/accounts/account123/action/interFranchiseMigration", "rel": "self" }
  ]
}
```

4.5.6 Requesting the status of a migration that does not exist

Request

GET {{APIBaseURL}}/accounts/account123/action/interFranchiseMigration

Response

HTTP/1.1 404 Not Found

```
{
  "itemNotFound" : {
    "message": "Resource not found",
    "code": 404,
    "details": "",
    "guid": "59e0b3dd-2173-432a-8ff5-60efe16e83bf"
  }
}
```

4.6 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

5 Accounts-accountId-action:POST

6 POST /accounts/{accountId}/action

Initiate an action on an account

6.1 Contents

- 1 POST /accounts/{accountId}/action
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Migrate an account to franchise "franchise-2" and set the new partner account Id to "partner1"
 - ◆ 1.4 See also

6.2 Request

POST /accounts/{accountId}/action

6.2.1 Parameters

accountId - *string*
The Id of the account.

6.2.2 URI Parameters

None.

6.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The *Authentication* credentials of the client application.

6.2.4 Request Body

```
{
  "action": "{action}",
  "actionData": {actionData}
}
```

6.2.4.1 Parameters

action - *string*
The action that should be performed. Allowed values are:
interFranchiseMigration - Submits a request to migrate this account to another franchise.

actionData - *dictionary*
Optional. Reserved for data specific for the requested action.

6.3 Response

6.3.1 Status Code

202 Accepted
The request has been successfully initiated. Check the action status for future updates.

400 Bad Request
The request body is not formatted correctly, is missing a required field, or contains an invalid value.

403 Forbidden
Forbidden access to users that are not authorized for such access.

404 Not Found
The account Id was not found.

409 Conflict
An action has already been submitted for this account and is in progress.

6.3.2 Response Body

None

6.4 Examples

6.4.1 Migrate an account to franchise "franchise-2" and set the new partner account Id to "partner1"

Request

```
POST {{APIBaseURL}}/accounts/account123/action
Content-Type:application/json; Charset=UTF-8
```

```
{
  action: "interFranchiseMigration",
  actionData: {
    franchise: "franchise-2",
    partnerAccountId: "partner1"
  }
}
```

Response

HTTP/1.1 202 Accepted

6.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

7 Accounts-accountId-attributes-attribute:DELETE

8 DELETE /accounts/{accountId}/attributes/{attribute}

Updates attribute value.

8.1 Contents

- 1 DELETE /accounts/{accountId}/attributes/{attribute}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Failure updating not existing attribute
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

8.2 Request

```
DELETE /accounts/{accountId}/attributes/{attribute}
```

8.2.1 Parameters

accountId - *string*
The Id of the Hostway account.

8.2.2 URI Parameters

attribute - *string*

8.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

Content-Type
application/json

8.3 Response

8.3.1 Status Code

204 No Content
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

8.4 Examples

8.4.1 Sending unauthorized request

Request

```
DELETE /accounts/account-id/attributes/test_attribute
```

Response

```
401 Unauthorized
```

8.4.2 Sending request without setting Content-Type

Request

```
DELETE /accounts/account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
```

Response

8.4.3 Failure updating not existing attribute

Request

```
DELETE /accounts/account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

404 Not Found

8.4.4 Success scenario

Request

```
DELETE /accounts/account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

204 No Content

8.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

9 Accounts-accountId-attributes-attribute:GET

10 GET /accounts/{accountId}/attributes/{attribute}

Get attribute data.

10.1 Contents

- 1 GET /accounts/{accountId}/attributes/{attribute}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Requesting not existing attribute
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

10.2 Request

GET /accounts/{accountId}/attributes/{attribute}

10.2.1 Parameters

accountId - *string*
The Id of the account.

10.2.2 URI Parameters

attribute - *string*

10.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The *Authentication* credentials of the client application.

Content-Type
application/json

10.3 Response

10.3.1 Status Code

200 OK
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

10.3.2 Response Body

```
{
  "name": "{name}",
  "value": {value},
  "links": [
    {
      "location": "{APIBaseURL}/accounts/{accountId}/attributes/{attribute}/",
      "rel": "self"
    }
  ]
}
```

10.3.2.1 Parameters

name - *string*
Descriptive name of the attribute

value - *string*
Value of the attribute

links - *list*
[Hypermedia](#) for this resource. Links to self.

10.4 Examples

10.4.1 Sending unauthorized request

Request

```
GET /accounts/account-id/attributes/test_attribute
```

Response

```
401 Unauthorized
```

10.4.2 Sending request without setting Content-Type

Request

```
GET /accounts/account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
```

Response

```
415 Unsupported Media Type
```

10.4.3 Requesting not existing attribute

Request

```
GET /accounts/account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
404 Not Found
```

10.4.4 Success scenario

Request

```
GET /accounts/account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
200 OK
```

```
{
  "name": "test_attribute",
  "value": "123",
  "links": [{"href": "{APIBaseURL}/accounts/account-id/attributes/test_attribute/", "rel": "related"}]
}
```

10.5 See also

- [Partner Account Attributes Management](#)
- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

11 Accounts-accountId-attributes-attribute:PUT

12 PUT /accounts/{accountId}/attributes/{attribute}

Updates attribute value.

12.1 Contents

- 1 PUT /accounts/{accountId}/attributes/{attribute}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Failure updating not existing attribute
 - ◇ 1.3.4 Failure updating attribute with missing required value
 - ◇ 1.3.5 Success scenario
 - ◆ 1.4 See also

12.2 Request

```
PUT /accounts/{accountId}/attributes/{attribute}
```

12.2.1 Parameters

accountId - *string*
The Id of the Hostway account.

12.2.2 URI Parameters

attribute - *string*

12.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

Content-Type
application/json

12.2.4 Request Body

```
{  
  "value": "{value}"  
}
```

12.2.4.1 Parameters

value - *string*

12.3 Response

12.3.1 Status Code

204 No Content
Success.

400 Bad Request
The format of the request body is invalid.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

12.4 Examples

12.4.1 Sending unauthorized request

Request

```
PUT /accounts/account-id/attributes/test_attribute
{"value": "test"}
```

Response

```
401 Unauthorized
```

12.4.2 Sending request without setting Content-Type

Request

```
PUT /accounts/account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
{"value": "test"}
```

Response

```
415 Unsupported Media Type
```

12.4.3 Failure updating not existing attribute

Request

```
PUT /accounts/account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"value": "test"}
```

Response

```
404 Not Found
```

12.4.4 Failure updating attribute with missing required value

Request

```
PUT /accounts/account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{}
```

Response

```
400 Bad Request
{"value": "Required"}
```

12.4.5 Success scenario

Request

```
PUT /accounts/account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"value": "test"}
```

Response

```
204 No Content
```

12.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

13 Accounts-accountId-attributes:GET

14 GET /accounts/{accountId}/attributes

Get attributes list.

14.1 Contents

- 1 GET /accounts/{accountId}/attributes
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◇ 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Getting list of attributes of not existing account
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

14.2 Request

GET /accounts/{accountId}/attributes

14.2.1 Parameters

accountId - *string*
The Id of the Hostway account.

14.2.2 URI Parameters

None.

14.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

Content-Type
application/json

14.3 Response

14.3.1 Status Code

200 OK
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

14.3.2 Response Body

```
{
  "list": [
    {
      "name": "{name}",
      "value": "{value}",
      "links": [{"href": "https://api.hostway.com/accounts/{accountId}/attributes/{name}", "rel": "related"}]
    },
    {
      "links": [
        {"href": "https://api.hostway.com/accounts/{accountId}/attributes", "rel": "self"}
      ]
    }
  ]
}
```

14.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about specific attribute

name - *string*
Descriptive name of the attribute

value - *string*
Value of the attribute

NOTE: The "partner_account_id" attribute will not be returned in this list.

links - *list*
[Hypermedia](#) for this product resource. Links to the attribute.

links - *list*
[Hypermedia](#) for this resource. Links to self.

14.4 Examples

14.4.1 Sending unauthorized request

Request

```
GET /accounts/account-id/attributes
```

Response

```
401 Unauthorized
```

14.4.2 Sending request without setting Content-Type

Request

```
GET /accounts/account-id/attributes
Authorization: Basic base64_encode(username:password)
```

Response

```
415 Unsupported Media Type
```

14.4.3 Getting list of attributes of not existing account

Request

```
GET /accounts/not-existing-account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
404 Not Found
```

14.4.4 Success scenario

Request

```
GET /accounts/account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
200 OK
```

```
{
  "list": [
    {
      "name": "test_attribute",
      "value": "123",
      "links": [{"href": "{{APIBaseURL}}/accounts/account-id/attributes/test_attribute/", "rel": "related"}]
    }
  ],
  "links": [
    {
      "href": "{{APIBaseURL}}/accounts/account-id/attributes/",
      "rel": "self"
    }
  ]
}
```

14.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

15 Accounts-accountId-attributes:POST

16 POST /accounts/{accountId}/attributes

Get attributes list.

16.1 Contents

- 1 POST /accounts/{accountId}/attributes
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Failure creating attribute for not existing account
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

16.2 Request

POST /accounts/{accountId}/attributes

16.2.1 Parameters

accountId - *string*
The Id of the Hostway account.

16.2.2 URI Parameters

None.

16.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

Content-Type
application/json

16.2.4 Request Body

```
{  
  "name": "test_attribute",  
  "value": "value"  
}
```

16.2.4.1 Parameters

name - *string*
Descriptive name of the attribute

value - *dictionary*
Value of the attribute

16.3 Response

16.3.1 Status Code

201 Created
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

16.4 Examples

16.4.1 Sending unauthorized request

Request

```
POST /accounts/account-id/attributes
{"name": "test_Account", "value": "test"}
```

Response

```
401 Unauthorized
```

16.4.2 Sending request without setting Content-Type

Request

```
POST /accounts/account-id/attributes
Authorization: Basic base64_encode(username:password)
{"name": "test_Account", "value": "test"}
```

Response

```
415 Unsupported Media Type
```

16.4.3 Failure creating attribute for not existing account

Request

```
POST /accounts/not-existing-account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"name": "test_Account", "value": "test"}
```

Response

```
404 Not Found
```

16.4.4 Success scenario

Request

```
POST /accounts/account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"name": "test_Account", "value": "test"}
```

Response

```
201 Created
```

16.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

17 Accounts-accountId-Contacts-contactType:GET

18 GET /accounts/{accountId}/contacts/{contactType}

Get specific contact type for an account

18.1 Contents

- 1 GET /accounts/{accountId}/contacts/{contactType}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Successful Request
 - ◇ 1.3.2 Request for non-existent contact type
 - ◆ 1.4 See also

18.2 Request

GET /accounts/{accountId}/contacts/{contactType}

18.2.1 Request Parameters

accountId - *string*

The unique account Id.

contactType - *string*

Contact type. One of 'regular', 'billing', 'administrator', 'technical'

18.2.2 URI Parameters

None.

18.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]

The *Authentication* credentials of the client application.

18.3 Response

18.3.1 Status Code

- 200 OK
Success
- 401 Unauthorized
The request was sent without an *Authorization* header, or the user represented by the value of this header was not properly authenticated.
- 403 Forbidden
The user authenticated for this request is not allowed to access the resource.
- 404 Not Found
Contact type does not match allowed values or the specified account Id does not exist or is purged.

18.3.2 Response Body

```
{
  "contactInfo": {
    "name": {
      "salutation": "{salutation}",
      "firstName": "{firstname}",
      "middleName": "{middleName}",
      "lastName": "{lastName}",
      "company": "{company}"
    },
    "address": {
      "street1": "{street1}",
      "street2": "{street2}",
      "city": "{city}",
      "stateOrProvince": "{stateOrProvince}",
      "postalCode": "{postalCode}",
      "countryCode": "{countryCode}"
    },
    "contactMedia": {
      "phone1": "{phone1}",
      "phone2": "{phone2}",
      "fax": "{fax}",
      "email1": "{email1}",
      "email2": "{email2}",
      "emailVerified": {emailverified}
    }
  }
}
```

```

    }
  },
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{accountId}/contacts/{contactType}/",
      "rel": "self"
    }
  ]
}

```

18.3.2.1 Parameters

city - *string*

May be up to 50 characters long. May contain only alphanumeric, white space and the following characters: -.'

company - *string*

(OPTIONAL)

May be up to 200 characters long. May contain only alphanumeric, white space and the following characters: -.'

countryCode - *string*

ISO 3166-1 alpha-2[2] country code.

email1 - *string*

May be up to 100 characters long. Must contain a valid e-mail address.

email2 - *string*

(OPTIONAL)

May be up to 100 characters long. If provided, must contain a valid e-mail address.

emailVerified - *integer*

Provides information whether the **email** is verified for a Zendesk access. Valid values are **0** or **1**.

fax - *string*

(OPTIONAL)

May be up to 20 characters long. If provided, must contain ITU-T E.164[3] formatted phone number.

firstName - *string*

May be up to 50 characters long. May contain only alphanumeric, white space and the following characters: -.'

lastName - *string*

May be up to 50 characters long. May contain only alphanumeric, white space and the following characters: -.'

middleName - *string*

(OPTIONAL)

May be up to 50 characters long. If provided, may contain only alphanumeric, white space and the following characters: -.'

phone1 - *string*

May be up to 20 characters long. Must contain ITU-T E.164[4] formatted phone number.

phone2 - *string*

(OPTIONAL)

May be up to 20 characters long. If provided, must contain ITU-T E.164[5] formatted phone number.

postalCode - *string*

(OPTIONAL)

May be up to 30 characters long. May contain only alphanumeric, white space and the following characters: -.' This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *postalCode* must be in format "#####" or "#####-####" (where "#" represents any numeric digit). If the value of *countryCode* is "CA", then *postalCode* must be in format "A#A#A#" or "A#A-#A#" or "A#A#A#" (where "A" represents any latin alphabet character and "#" represents any numeric digit).

salutation - *string*

(OPTIONAL)

May contain one of the salutations provided by [GET /salutations](#)

stateOrProvince - *string*

(OPTIONAL)

May be up to 20 characters long. May contain only alphanumeric, white space and the following characters: -.' This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *stateOrProvince* must be a valid two-letter US state abbreviation. If the value of *countryCode* is "CA", then *stateOrProvince* must be a valid two-letter Canadian province abbreviation.

street1 - *string*

May be up to 100 characters long. May contain only alphanumeric, white space and the following characters: -.'

street2 - *string*

(OPTIONAL)

May be up to 100 characters long. If provided, may contain only alphanumeric, white space and the following characters: -.'

18.4 Examples

18.4.1 Successful Request

Request

```

GET /accounts/{accountId}/contacts/regular
Authorization: Basic {Base64-encoded username & password string}

```

Response

200 OK

```

{
  "contactInfo": {
    "name": {
      "salutation": "Mr.",
      "firstName": "John",
      "middleName": "",
      "lastName": "Smith",
      "company": "TestCo, Inc."
    },
    "address": {
      "street1": "100 Main St.",
      "street2": "",
      "city": "Chicago",
      "stateOrProvince": "IL",
      "postalCode": "60001",
      "countryCode": "US"
    },
    "contactMedia": {
      "phone1": "555-555-5501",
      "phone2": "",
      "fax": "",
      "email1": "test@test.com",
      "email2": "",
      "emailVerified": 1
    }
  },
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{accountId}/contacts/regular/",
      "rel": "self"
    }
  ]
}

```

18.4.2 Request for non-existent contact type

Request

```

GET /accounts/{accountId}/contacts/fake
Authorization: Basic {Base64-encoded username & password string}

```

Response

404 Not Found

```

{
  "itemNotFound": {
    "guid": {identifier},
    "message": "Resource not found",
    "code": 404,
    "details": ""
  }
}

```

18.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

19 Accounts-accountId-Contacts-contactType:PUT

20 PUT /accounts/{accountId}/contacts/{contactType}

Set specific contact type for an account

20.1 Contents

- 1 PUT /accounts/{accountId}/contacts/{contactType}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Successful update of contact information
 - ◇ 1.3.2 Request for updating non-existent contact type
 - ◇ 1.3.3 Request for updating contact type with not valid data
 - ◇ 1.3.4 Request for updating contact type with missing or empty required data
 - ◆ 1.4 See also

20.2 Request

PUT /accounts/{accountId}/contacts/{contactType}

20.2.1 Parameters

accountId - *string*
The Id of the account.

contactType - *string*
Contact type. One of 'regular', 'billing', 'administrator', 'technical'

20.2.2 URI Parameters

None.

20.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The **Authentication** credentials of the client application.

20.2.4 Request Body

```
{
  "name": {
    "salutation": "{salutation}",
    "firstName": "{firstname}",
    "middleName": "{middleName}",
    "lastName": "{lastName}",
    "company": "{company}"
  },
  "address": {
    "street1": "{street1}",
    "street2": "{street2}",
    "city": "{city}",
    "stateOrProvince": "{stateOrProvince}",
    "postalCode": "{postalCode}",
    "countryCode": "{countryCode}"
  },
  "contactMedia": {
    "phone1": "{phone1}",
    "phone2": "{phone2}",
    "fax": "{fax}",
    "email1": "{email1}",
    "email2": "{email2}"
  }
}
```

20.2.4.1 Parameters

city - *string*
(**REQUIRED**)
May be up to 50 characters long. May contain only alphanumeric, white space and the following characters: -'

company - *string*
(**OPTIONAL**)
May be up to 200 characters long. May contain only alphanumeric, white space and the following characters: -'

countryCode - *string*
(**REQUIRED**)
ISO 3166-1 alpha-2[2] country code.

email1 - *string*
(**REQUIRED**)

May be up to 100 characters long. Must contain a valid e-mail address.

email2 - *string*
(OPTIONAL)

May be up to 100 characters long. If provided, must contain a valid e-mail address.

fax - *string*
(OPTIONAL)

May be up to 20 characters long. If provided, must contain ITU-T E.164[3] formatted phone number.

firstName - *string*
(REQUIRED)

May be up to 50 characters long. May contain only alphanumeric, white space and the following characters: -.'

lastName - *string*
(REQUIRED)

May be up to 50 characters long. May contain only alphanumeric, white space and the following characters: -.'

middleName - *string*
(OPTIONAL)

May be up to 50 characters long. If provided, may contain only alphanumeric, white space and the following characters: -.'

phone1 - *string*
(REQUIRED)

May be up to 20 characters long. Must contain ITU-T E.164[4] formatted phone number.

phone2 - *string*
(OPTIONAL)

May be up to 20 characters long. If provided, must contain ITU-T E.164[5] formatted phone number.

postalCode - *string*
(OPTIONAL)

May be up to 30 characters long. May contain only alphanumeric, white space and the following characters: -.' This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *postalCode* must be in format "#####" or "#####-####" (where "#" represents any numeric digit). If the value of *countryCode* is "CA", then *postalCode* must be in format "A#A#A#" or "A#A-#A#" or "A#A#A#" (where "A" represents any latin alphabet character and "#" represents any numeric digit).

salutation - *string*
(OPTIONAL)

May contain one of the salutations provided by [GET /salutations](#)

stateOrProvince - *string*
(OPTIONAL)

May be up to 20 characters long. May contain only alphanumeric, white space and the following characters: -.' This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *stateOrProvince* must be a valid two-letter US state abbreviation. If the value of *countryCode* is "CA", then *stateOrProvince* must be a valid two-letter Canadian province abbreviation.

street1 - *string*
(REQUIRED)

May be up to 100 characters long. May contain only alphanumeric, white space and the following characters: -.'

street2 - *string*
(OPTIONAL)

May be up to 100 characters long. If provided, may contain only alphanumerics, white space and the following characters: -.'

20.3 Response

20.3.1 Status Code

204 No Content

Successfully updated contact information

400 Bad Request

Contact information was not updated because either the request body was malformed, or some element of the request body did not meet the requirements.

401 Unauthorized

The request was sent without an *Authorization* header, or the user represented by the value of this header was not properly authenticated.

403 Forbidden

The user authenticated for this request is not allowed to access the resource.

404 Not Found

The account Id was not found.

20.3.2 Response Body

None.

20.4 Examples

20.4.1 Successful update of contact information

Request

PUT /accounts/{accountId}/contacts/regular

Authorization: Basic {Base64-encoded username & password string}

```
{
  "name": {
    "salutation": "Mr.",
    "firstName": "John",
    "middleName": "",
    "lastName": "Smith",
    "company": "TestCo, Inc."
  },
  "address": {
    "street1": "100 Main St.",
    "street2": "",
    "city": "Chicago",
    "stateOrProvince": "IL",
    "postalCode": "60001",
    "countryCode": "US"
  },
  "contactMedia": {
    "phone1": "555-555-5501",
    "phone2": "",
    "fax": "",
    "email1": "test@test.com",
    "email2": ""
  }
}
```

Response

204 No Content

20.4.2 Request for updating non-existent contact type

Request

PUT /accounts/{accountId}/contacts/fake

Authorization: Basic {Base64-encoded username & password string}

```
{
  "name": {
    "salutation": "Mr.",
    "firstName": "John",
    "middleName": "",
    "lastName": "Smith",
    "company": "TestCo, Inc."
  },
  "address": {
    "street1": "100 Main St.",
    "street2": "",
    "city": "Chicago",
    "stateOrProvince": "IL",
    "postalCode": "60001",
    "countryCode": "US"
  },
  "contactMedia": {
    "phone1": "555-555-5501",
    "phone2": "",
    "fax": "",
    "email1": "test@test.com",
    "email2": ""
  }
}
```

Response

404 Not Found

```
{
  "itemNotFound": {
    "guid": {identifier},
    "message": "Resource not found",
    "code": 404,
    "details": ""
  }
}
```

20.4.3 Request for updating contact type with not valid data

Request

PUT /accounts/{accountId}/contacts/regular

Authorization: Basic {Base64-encoded username & password string}

```
{
  "name": {
    "salutation": "Mr.",
    "firstName": "John",
    "middleName": "",
    "lastName": "Smith",
    "company": "TestCo, Inc."
  },
}
```

```

"address": {
  "street1": "100 Main St.",
  "street2": "",
  "city": "Chicago",
  "stateOrProvince": "IL",
  "postalCode": "60001",
  "countryCode": "USA"
},
"contactMedia": {
  "phone1": "555-555-5501",
  "phone2": "",
  "fax": "",
  "email1": "test@test.com",
  "email2": ""
}
}

```

Response

400 Bad Request

```

{
  "badRequest": {
    "guid": {identifier},
    "message": "POST data error",
    "code": 400,
    "details": {
      "countryCode": "Invalid Country Code"
    }
  }
}

```

20.4.4 Request for updating contact type with missing or empty required data

Request

PUT /accounts/{accountId}/contacts/regular

Authorization: Basic {Base64-encoded username & password string}

```

{
  "name": {
    "salutation": "Mr.",
    "firstName": "",
    "middleName": "",
    "lastName": "Smith",
    "company": "TestCo, Inc."
  },
  "address": {
    "street1": "100 Main St.",
    "street2": "",
    "city": "Chicago",
    "stateOrProvince": "IL",
    "postalCode": "60001"
  },
  "contactMedia": {
    "phone1": "555-555-5501",
    "phone2": "",
    "fax": "",
    "email1": "test@test.com",
    "email2": ""
  }
}

```

Response

400 Bad Request

```

{
  "badRequest": {
    "guid": {identifier},
    "message": "POST data error",
    "code": 400,
    "details": {
      "countryCode": "Required",
      "firstName": "Required"
    }
  }
}

```

20.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

21 Accounts-accountId-Contacts:GET

22 GET /accounts/{accountId}/contacts

Get contact details for an account.

22.1 Contents

- 1 GET /accounts/{accountId}/contacts
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Successful Request
 - ◇ 1.3.2 Request for non-existent account Id
 - ◆ 1.4 See also

22.2 Request

GET /accounts/{accountId}/contacts

22.2.1 Request Parameters

accountId - *string*
The unique account Id.

22.2.2 URI Parameters

None.

22.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

22.3 Response

22.3.1 Status Code

- 200 OK
Success
- 401 Unauthorized
The request was sent without an *Authorization* header, or the user represented by the value of this header was not properly authenticated.
- 403 Forbidden
The user authenticated for this request is not allowed to access the resource.
- 404 Not Found
The specified account Id does not exist or is purged.

22.3.2 Response Body

```
{
  "contactInfo": {
    "regular": {
      "name": {
        "salutation": "{salutation}",
        "firstName": "{firstname}",
        "middleName": "{middleName}",
        "lastName": "{lastName}",
        "company": "{company}"
      },
      "address": {
        "street1": "{street1}",
        "street2": "{street2}",
        "city": "{city}",
        "stateOrProvince": "{stateOrProvince}",
        "postalCode": "{postalCode}",
        "countryCode": "{countryCode}"
      }
    }
  }
}
```

```

    },
    "contactMedia": {
      "phone1": "{phone1}",
      "phone2": "{phone2}",
      "fax": "{fax}",
      "email1": "{email1}",
      "email2": "{email2}",
      "emailVerified": {emailverified}
    }
  },
  "billing": {
    ...
  },
  "administrator": {
    ...
  },
  "technical": {
    ...
  }
},
"links": [
  {
    "href": "https://api.hostway.com/accounts/{accountId}/contacts/",
    "rel": "self"
  }
]
}

```

22.3.2.1 Parameters

Note, there are 4 types of contact information that must be returned with the response: "regular", "billing", "administrator" and "technical".

city - *string*

May be up to 50 characters long. May contain only alphanumerics, white space and the following characters: -'

company - *string*

(OPTIONAL)

May be up to 200 characters long. May contain only alphanumerics, white space and the following characters: -'

countryCode - *string*

ISO 3166-1 alpha-2[2] country code.

email1 - *string*

May be up to 100 characters long. Must contain a valid e-mail address.

email2 - *string*

(OPTIONAL)

May be up to 100 characters long. If provided, must contain a valid e-mail address.

emailVerified - *integer*

Provides information whether the **email** is verified for a Zendesk access. Valid values are **0** or **1**.

fax - *string*

(OPTIONAL)

May be up to 20 characters long. If provided, must contain ITU-T E.164[3] formatted phone number.

firstName - *string*

May be up to 50 characters long. May contain only alphanumerics, white space and the following characters: -'

lastName - *string*

May be up to 50 characters long. May contain only alphanumerics, white space and the following characters: -'

middleName - *string*

(OPTIONAL)

May be up to 50 characters long. If provided, may contain only alphanumerics, white space and the following characters: -'

phone1 - *string*

May be up to 20 characters long. Must contain ITU-T E.164[4] formatted phone number.

phone2 - *string*

(OPTIONAL)

May be up to 20 characters long. If provided, must contain ITU-T E.164[5] formatted phone number.

postalCode - *string*

(OPTIONAL)

May be up to 30 characters long. May contain only alphanumerics, white space and the following characters: -'. This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *postalCode* must be in format "#####" or "#####-####" (where "#" represents any numeric digit). If the value of *countryCode* is "CA", then *postalCode* must be in format "A#A#A#" or "A#A-#A#" or "A#A #A#" (where "A" represents any latin alphabet character and "#" represents any numeric digit).

salutation - *string*

(OPTIONAL)

May contain one of the salutations provided by [GET /salutations](#)

stateOrProvince - *string*

(OPTIONAL)

May be up to 20 characters long. May contain only alphanumerics, white space and the following characters: -'. This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *stateOrProvince* must be a valid two-letter US state abbreviation. If the value of *countryCode* is "CA", then *stateOrProvince* must be a valid two-letter Canadian province abbreviation.

street1 - *string*

May be up to 100 characters long. May contain only alphanumerics, white space and the following characters: -'

street2 - string
(OPTIONAL)

May be up to 100 characters long. If provided, may contain only alphanumerics, white space and the following characters: -.'

22.4 Examples

22.4.1 Successful Request

Request

```
GET /accounts/{accountId}/contacts
Authorization: Basic {Base64-encoded username & password string}
```

Response

200 OK

```
{
  "contactInfo": {
    "regular": {
      "name": {
        "salutation": "Mr.",
        "firstName": "John",
        "middleName": "",
        "lastName": "Smith",
        "company": "TestCo, Inc."
      },
      "address": {
        "street1": "100 Main St.",
        "street2": "",
        "city": "Chicago",
        "stateOrProvince": "IL",
        "postalCode": "60001",
        "countryCode": "US"
      },
      "contactMedia": {
        "phone1": "555-555-5501",
        "phone2": "",
        "fax": "",
        "email1": "test@test.com",
        "email2": "",
        "emailVerified": 1
      }
    },
    "billing": {
      "name": {
        "salutation": "Mr.",
        "firstName": "John",
        "middleName": "",
        "lastName": "Smith",
        "company": "TestCo, Inc."
      },
      "address": {
        "street1": "100 Main St.",
        "street2": "",
        "city": "Chicago",
        "stateOrProvince": "IL",
        "postalCode": "60001",
        "countryCode": "US"
      },
      "contactMedia": {
        "phone1": "555-555-5501",
        "phone2": "",
        "fax": "",
        "email1": "test@test.com",
        "email2": "",
        "emailVerified": 1
      }
    },
    "administrator": {
      "name": {
        "salutation": "Mr.",
        "firstName": "John",
        "middleName": "",
        "lastName": "Smith",
        "company": "TestCo, Inc."
      },
      "address": {
        "street1": "100 Main St.",
        "street2": "",
        "city": "Chicago",
        "stateOrProvince": "IL",
        "postalCode": "60001",
        "countryCode": "US"
      },
      "contactMedia": {
        "phone1": "555-555-5501",
        "phone2": "",
        "fax": "",
        "email1": "test@test.com",
        "email2": "",
        "emailVerified": 1
      }
    },
    "technical": {
      "name": {
        "salutation": "Mr.",
        "firstName": "John",
        "middleName": "",
        "lastName": "Smith",
        "company": "TestCo, Inc."
      },
      "address": {
```

```

    "street1": "100 Main St.",
    "street2": "",
    "city": "Chicago",
    "stateOrProvince": "IL",
    "postalCode": "60001",
    "countryCode": "US"
  },
  "contactMedia": {
    "phone1": "555-555-5501",
    "phone2": "",
    "fax": "",
    "email1": "test@test.com",
    "email2": "",
    "emailVerified": 1
  }
},
"links": [
  {
    "href": "https://api.hostway.com/accounts/{accountId}/contacts/",
    "rel": "self"
  }
]
}

```

22.4.2 Request for non-existent account Id

Request

```

GET /accounts/{fake_account_number}/contacts
Authorization: Basic {Base64-encoded username & password string}

```

Response

```

404 Not Found

{
  "itemNotFound": {
    "guid": {identifier},
    "message": "/accounts/fake-account-id/contacts",
    "code": 404,
    "details": ""
  }
}

```

22.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

23 Accounts-accountId-Contacts:PUT

24 PUT /accounts/{accountId}/contacts

Set the contact details for an account

24.1 Contents

- 1 PUT /accounts/{accountId}/contacts
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Successful update of contact information
 - ◆ 1.4 See also

24.2 Request

PUT /accounts/{accountId}/contacts

24.2.1 Parameters

accountId - *string*
The Id of the account.

24.2.2 URI Parameters

None.

24.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The **Authentication** credentials of the client application.

24.2.4 Request Body

```
{
  "regular": {
    "name": {
      "salutation": "{salutation}",
      "firstName": "{firstname}",
      "middleName": "{middleName}",
      "lastName": "{lastName}",
      "company": "{company}"
    },
    "address": {
      "street1": "{street1}",
      "street2": "{street2}",
      "city": "{city}",
      "stateOrProvince": "{stateOrProvince}",
      "postalCode": "{postalCode}",
      "countryCode": "{countryCode}"
    },
    "contactMedia": {
      "phone1": "{phone1}",
      "phone2": "{phone2}",
      "fax": "{fax}",
      "email1": "{email1}",
      "email2": "{email2}"
    }
  },
  "billing": {
    ...
  },
  "administrator": {
    ...
  },
  "technical": {
    ...
  }
}
```

24.2.4.1 Parameters

Note, there are 4 types of contact information that must be sent with the request: "regular", "billing", "administrator" and "technical". All 4 are required.

city - *string*
(REQUIRED)
May be up to 50 characters long. May contain only alphanumerics, white space and the following characters: -'

company - *string*
(*OPTIONAL*)
May be up to 200 characters long. May contain only alphanumerics, white space and the following characters: -.'

countryCode - *string*
(*REQUIRED*)
ISO 3166-1 alpha-2[2] country code.

email1 - *string*
(*REQUIRED*)
May be up to 100 characters long. Must contain a valid e-mail address.

email2 - *string*
(*OPTIONAL*)
May be up to 100 characters long. If provided, must contain a valid e-mail address.

fax - *string*
(*OPTIONAL*)
May be up to 20 characters long. If provided, must contain ITU-T E.164[3] formatted phone number.

firstName - *string*
(*REQUIRED*)
May be up to 50 characters long. May contain only alphanumerics, white space and the following characters: -.'

lastName - *string*
(*REQUIRED*)
May be up to 50 characters long. May contain only alphanumerics, white space and the following characters: -.'

middleName - *string*
(*OPTIONAL*)
May be up to 50 characters long. If provided, may contain only alphanumerics, white space and the following characters: -.'

phone1 - *string*
(*REQUIRED*)
May be up to 20 characters long. Must contain ITU-T E.164[4] formatted phone number.

phone2 - *string*
(*OPTIONAL*)
May be up to 20 characters long. If provided, must contain ITU-T E.164[5] formatted phone number.

postalCode - *string*
(*OPTIONAL*)
May be up to 30 characters long. May contain only alphanumerics, white space and the following characters: -.' This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *postalCode* must be in format "#####" or "#####-####" (where "#" represents any numeric digit). If the value of *countryCode* is "CA", then *postalCode* must be in format "A#A#A#" or "A#A-#A#" or "A#A#A#" (where "A" represents any latin alphabet character and "#" represents any numeric digit).

salutation - *string*
(*OPTIONAL*)
May contain one of the salutations provided by [GET /salutations](#)

stateOrProvince - *string*
(*OPTIONAL*)
May be up to 20 characters long. May contain only alphanumerics, white space and the following characters: -.' This value is *required* if the value of *countryCode* is "US" or "CA". If the value of *countryCode* is "US", then *stateOrProvince* must be a valid two-letter US state abbreviation. If the value of *countryCode* is "CA", then *stateOrProvince* must be a valid two-letter Canadian province abbreviation.

street1 - *string*
(*REQUIRED*)
May be up to 100 characters long. May contain only alphanumerics, white space and the following characters: -.'

street2 - *string*
(*OPTIONAL*)
May be up to 100 characters long. If provided, may contain only alphanumerics, white space and the following characters: -.'

24.3 Response

24.3.1 Status Code

204 No Content
Successfully updated contact information

400 Bad Request
Contact information was not updated because either the request body was malformed, or some element of the request body did not meet the requirements.

401 Unauthorized
The request was sent without an *Authorization* header, or the user represented by the value of this header was not properly authenticated.

403 Forbidden
The user authenticated for this request is not allowed to access the resource.

404 Not Found
The account Id was not found.

24.3.2 Response Body

None.

24.4 Examples

24.4.1 Successful update of contact information

Request

```
PUT /accounts/{accountId}/contacts
```

```
Authorization: Basic {Base64-encoded username & password string}
```

```
{
  "regular": {
    name: {
      "salutation": "Mr.",
      "firstName": "John",
      "middleName": "",
      "lastName": "Smith",
      "company": "TestCo, Inc."
    },
    "address": {
      "street1": "100 Main St.",
      "street2": "",
      "city": "Chicago",
      "stateOrProvince": "IL",
      "postalCode": "60001",
      "countryCode": "US"
    },
    "contactMedia": {
      "phone1": "555-555-5501",
      "phone2": "",
      "fax": "",
      "email1": "test@test.com",
      "email2": ""
    }
  },
  "billing": {
    name: {
      "salutation": "Mr.",
      "firstName": "John",
      "middleName": "",
      "lastName": "Smith",
      "company": "TestCo, Inc."
    },
    "address": {
      "street1": "100 Main St.",
      "street2": "",
      "city": "Chicago",
      "stateOrProvince": "IL",
      "postalCode": "60001",
      "countryCode": "US"
    },
    "contactMedia": {
      "phone1": "555-555-5501",
      "phone2": "",
      "fax": "",
      "email1": "test@test.com",
      "email2": ""
    }
  },
  "administrator": {
    name: {
      "salutation": "Mr.",
      "firstName": "John",
      "middleName": "",
      "lastName": "Smith",
      "company": "TestCo, Inc."
    },
    "address": {
      "street1": "100 Main St.",
      "street2": "",
      "city": "Chicago",
      "stateOrProvince": "IL",
      "postalCode": "60001",
      "countryCode": "US"
    },
    "contactMedia": {
      "phone1": "555-555-5501",
      "phone2": "",
      "fax": "",
      "email1": "test@test.com",
      "email2": ""
    }
  },
  "technical": {
    name: {
      "salutation": "Mr.",
      "firstName": "John",
      "middleName": "",
      "lastName": "Smith",
      "company": "TestCo, Inc."
    },
    "address": {
      "street1": "100 Main St.",
      "street2": "",
      "city": "Chicago",
      "stateOrProvince": "IL",
      "postalCode": "60001",
      "countryCode": "US"
    }
  }
}
```

```
    },
    "contactMedia": {
      "phone1": "555-555-5501",
      "phone2": "",
      "fax": "",
      "email1": "test@test.com",
      "email2": ""
    }
  }
}
```

Response

204 No Content

24.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

25 Accounts-accountId-discount:GET

26 GET /accounts/{accountId}/discount

Get account level discount

26.1 Contents

- 1 GET /accounts/{accountId}/discount
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Success scenario
 - ◆ 1.4 See also

26.2 Request

GET /accounts/{accountId}/discount

26.2.1 Request Parameters

accountId - *string*
The user account on which the discount is applied

26.3 Response

26.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

404 Not Found
The account or invoice does not exist.

26.3.2 Response Body

```
{
  "discountPercentage": "{discountPercentage}",
  "links":
  [
    {
      "href": "{APIBaseURL}/accounts/{accountId}/discount",
      "rel": "self"
    }
  ]
}
```

26.3.2.1 Parameters

discountPercentage - *integer*
Discount percentage

accountId - *string*
The user account on which the discount is applied

links - *string*
Hypermedia for this resource. Links to self

26.4 Examples

26.4.1 Success scenario

Request

GET /accounts/test-account/discount

Response

```
{
  {
    "discountPercentage": 30,
```

```
    "links":  
    [  
      {  
        "href": "http://{{APIBaseURL}}/accounts/test-account/discount/",  
        "rel": "self"  
      }  
    ]  
  }  
}
```

26.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

27 Accounts-accountId-invoices-invoiceId-transactions:GET

28 GET /accounts/{accountId}/invoices/{invoiceId}/transactions

Retrieves a list of transactions for a specific account and and invoice

28.1 Contents

- 1 GET /accounts/{accountId}/invoices/{invoiceId}/transactions
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Success scenario
 - ◆ 1.4 See also

28.2 Request

GET /accounts/{accountId}/invoices/{invoiceId}/transactions

28.2.1 Request Parameters

accountId - *string*
The user account owning the invoice

invoiceId - *string*
The specific invoice for which the call will retrieve the list

28.2.2 URI Parameters

filterFrom - *date*
Only transactions having startDate after the value of the parameter will be returned

filterTo - *date*
Only transactions having startDate before the value of the parameter will be returned

filterType - *string*
Only transactions having type matching the value of the parameter will be returned. Possible values are: 'charge', 'credit', 'discount'

page - *integer*
Specifies which page should be displayed. Requires a pageSize parameter to also be provided

pageSize - *integer*
Specifies the number of entries to be displayed on a page. Requires a page parameter to also be provided

sortField - *string*
Specifies if the resulting list is to be sorted by a given field. Allowed values are **amount**, **from**, **to**

sortOrder - *string*
The order in which the sorting is to be done. Allowed values are **asc** and **desc**

allowRollup - *boolean*
Allows roll up of any charge or credit transactions in an invoice if the product associated with the transaction supports roll up. Default is not to allow roll up.

28.3 Response

28.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

404 Not Found
The account or invoice does not exist.

28.3.2 Response Body

```
{
  "list": [
    {
      "startDate": "{startDate}",
      "endDate": {endDate},
      "currency": {currency},
      "amount": "{amount}",
      "description": "{description}",
      "date": "{date}",
      "productSpecification": "{productSpecification}",
      "serviceNumber": "{serviceNumber}",
      "type": "{type}",
      "number": "{number}",
      "isRolledUp": "{isRolledUp}"
    }
  ],
}
```

```

    "links": [
      {
        "href": "{href}",
        "rel": "{rel}"
      }
    ]
  }
}

```

28.3.2.1 Parameters

startDate - *string*
Start date of transaction

endDate - *string*
End date of transaction

currency - *string*
Currency of transaction

amount - *string*
Amount of transaction

description - *string*
Description of transaction

date - *string*
Date of transaction

productSpecification - *string*
Product specification of transaction

serviceNumber - *string*
Service number of transaction

type - *string*
Type of transaction. Possible values are: 'charge', 'credit', 'discount'

number - *string*
Number of transaction

isRolledUp - *integer*
Indicates if transaction is the aggregate of several rolled up transactions for the invoice. Possible values 1 if rolled up, 0 if not

href - *string*
Link to other resources relevant to the mailboxes lists

rel - *string*
Type of relation to the resource for the provided link

28.4 Examples

28.4.1 Success scenario

Request

```
GET /accounts/test-account/invoices/11111/transactions/
```

Response

```

{
  "list": [
    {
      "startDate": "2011-06-19",
      "amount": 34.65,
      "endDate": "2011-07-18",
      "description": "Usage Fee test@test.com Rate Type = Exchange mailbox usage, Usage amount = 7",
      "currency": "USD", "date": "2011-11-10",
      "productSpecification": "{APIBaseURL}/productSpecifications/test-account-000001311f19185dc1df5d10000a000a000800",
      "serviceNumber": "test-account-0000013128ec0f17330329ef000a000a000800",
      "type": "charge",
      "number": 101378490,
      "isRolledUp": 0
    },
    ...
  ],
  "links": [
    { "href": "{APIBaseURL}/accounts/test-account/invoices/11111/transactions?page=1&pageSize=100", "rel": "self" },
    { "href": "{APIBaseURL}/accounts/test-account/invoices/11111/transactions?page=2&pageSize=100", "rel": "next" },
    { "href": "{APIBaseURL}/accounts/test-account/invoices/11111/transactions?page=5&pageSize=100", "rel": "last" }
  ]
}

```

28.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

29 Accounts-accountId-invoicesInfo-filtered-status:GET

30 GET /accounts/{accountId}/invoice?filterStatus={filterStatus}

Gets a list of account's invoices filtered by status

30.1 Contents

- 1 GET /accounts/{accountId}/invoice?filterStatus={filterStatus}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET paginated invoices list call
 - ◆ 1.4 See also

30.2 Request

GET /accounts/{accountId}/invoices?filterStatus={filterStatus}

30.2.1 Parameters

accountId - *string*
The Id of the account.

filterStatus - *string*
Status of invoice. Possible values are: 'invoice_paid', 'invoice_unpaid', 'invoice_open'.

30.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

30.3 Response

30.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The accountId was not found.

30.3.2 Response Body

```
{
  "list": [
    {
      "status": "{status}",
      "currency": "{currency}",
      "amount": "{amount}",
      "amountDue": "{amountDue}",
      "description": "{description}",
      "date": "{date}",
      "statusName": "{statusName}",
      "invoiceNumber": "{invoiceNumber}"
    }
  ],
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{account_id}/invoices",
      "rel": "self"
    }
  ]
}
```

30.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about invoices

status - *string*
Status of invoice

currency - *string*
Currency of invoice

ammount - *string*
Amount of invoice

amountDue - *string*
Amount due of invoice

description - *string*
Description of invoice

date - *string*
Date of invoice

statusName - *string*
Status of invoice

invoiceNumber - *string*
Invoice number

links - *list*
[Hypermedia](#) for this resource. Links to self

30.4 Examples

30.4.1 GET paginated invoices list call

Request

GET `{{APIBaseURL}}/accounts/test731993/invoices?filterStatus=invoice_paid&page=1&pageSize=5`

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "status": "invoice_paid",
    "currency": "USD",
    "amount": "1023.7200",
    "amountDue": "0.0000",
    "description": "Invoice #SO-360704",
    "date": "2011-04-19",
    "statusName": "Invoice Paid",
    "invoiceNumber": "SO-360704"
  } ],
  "links" : [
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=1&pageSize=5",
      "rel" : "self"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=2&pageSize=5",
      "rel" : "next"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=44&pageSize=5",
      "rel" : "last"
    }
  ]
}
```

30.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

31 Accounts-accountId-invoicesInfo-filtered-type:GET

32 GET /accounts/{accountId}/invoice?filterStatus={filterStatus}

Gets a list of account's invoices filtered by status

32.1 Contents

- 1 GET
/accounts/{accountId}/invoice?filterStatus={filterStatus}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET paginated invoices list call
 - ◆ 1.4 See also

32.2 Request

GET /accounts/{accountId}/invoices?filterStatus={filterStatus}

32.2.1 Parameters

accountId - *string*
The Id of the account.

filterType - *string*
Type of transaction. Possible values are: 'charge', 'charge_back', 'payment', 'fee', 'refund', 'credit', 'cc_chargeback', 'charge_adjustment', 'discount', 'credit_adjustment', 'credit_proration', 'charge_proration', 'payment_adjustment', 'tax', 'cc_chargeback_reversal', 'usage_charge', 'write_off', 'undo_write_off', 'usage_discount'

32.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

32.3 Response

32.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The accountId was not found.

32.3.2 Response Body

```
{
  "list": [
    {
      "status": "{status}",
      "currency": "{currency}",
      "amount": "{amount}",
      "amountDue": "{amountDue}",
      "description": "{description}",
      "date": "{date}",
      "statusName": "{statusName}",
      "invoiceNumber": "{invoiceNumber}"
    }
  ],
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{account_id}/invoices",
      "rel": "self"
    }
  ]
}
```

32.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about invoices

status - *string*
Status of invoice

currency - *string*
Currency of invoice

ammount - *string*
Amount of invoice

amountDue - *string*
Amount due of invoice

description - *string*
Description of invoice

date - *string*
Date of invoice

statusName - *string*
Status of invoice

invoiceNumber - *string*
Invoice number

links - *list*
[Hypermedia](#) for this resource. Links to self

32.4 Examples

32.4.1 GET paginated invoices list call

Request

GET [{{APIBaseURL}}/accounts/test731993/invoices?filterStatus=invoice_paid&page=1&pageSize=5](#)

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "startDate": "2012-04-23",
    "amount": 0.87,
    "endDate": "2012-05-18",
    "description": "Recurring fee Exchange 2010 Archiving, core.email.exchange2010.archiving 04/23/2012 through 05/18/2012",
    "currency": "USD",
    "date": "2012-04-23",
    "productSpecification": "{{APIBaseURL}}/productSpecifications/test731993-00000137a32daee9e2b08715000a30fe33d32c",
    "serviceNumber": "test731993-00000136dedf26dfd9ec22fd000a1ddbba01da1",
    "type": "payment",
    "number": 101393524
  } ],
  "links" : [
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=1&pageSize=5",
      "rel" : "self"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=2&pageSize=5",
      "rel" : "next"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=44&pageSize=5",
      "rel" : "last"
    }
  ]
}
```

32.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

33 Accounts-accountId-invoicesInfo-invoiceId:GET

34 GET /accounts/{accountId}/invocest/{invoiceId}

Retrieves an individual invoice.

34.1 Contents

- 1 GET /accounts/{accountId}/invocest/{invoiceId}
- ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
- ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
- ◆ 1.3 Examples
 - ◇ 1.3.1 GET invoice
 - ◇ 1.3.2 GET invoice PDF
- ◆ 1.4 See also

34.2 Request

GET /accounts/{accountId}/invocest/{invoiceId}

34.2.1 Parameters

accountId - *string*
The Id of the account.

invoiceId - *string*
The Id of the invoice.

34.2.2 URI Parameters

None.

34.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]

The [Authentication](#) credentials of the client application.

Accept - *HTTP Accept header*. Possible values are *application/json*, *application/pdf*. If omitted *json* will be returned [2]

34.3 Response

34.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The account or invoice was not found

34.3.2 Response Body

```
{
  "status": "{status}",
  "currency": "{currency}",
  "amount": "{amount}",
  "amountDue": "{amountDue}",
  "description": "{description}",
  "links": [{
    "href": "https://api.hostway.com/accounts/{accountId}/invocest/{invoiceId}/",
    "rel": "self"
  }],
  "date": "{date}",
  "statusName": "{statusName}",
  "invoiceNumber": "{invoiceNumber}"
}
```

34.3.2.1 Parameters

status - *string*
Status of invoice

currency - *string*
Currency of invoice

ammount - *string*
Amount of invoice

amountDue - *string*
Amount due of invoice

description - *string*
Description of invoice

date - *string*
Date of invoice

statusName - *string*
Status of invoice

invoiceNumber - *string*
Invoice number

links - *list*
[Hypermedia](#) for this resource. Links to self

34.4 Examples

34.4.1 GET invoice

Request

GET [{{APIBaseURL}}/accounts/test731993/invoices/123456](#)

Response

HTTP/1.1 200 OK

```
{
  "status": "invoice_paid",
  "currency": "USD",
  "amount": "1113.7500",
  "amountDue": "0.0000",
  "description": "Invoice #123456",
  "links": [
    {
      "href": "{{APIBaseURL}}/accounts/test731993/invoices/123456/",
      "rel": "self"
    }
  ],
  "date": "2011-11-10",
  "statusName": "Invoice Paid",
  "invoiceNumber": "123456"
}
```

34.4.2 GET invoice PDF

Request

GET [{{APIBaseURL}}/accounts/test731993/invoices/123456](#) Accept: application/pdf

Response

HTTP/1.1 200 OK

Content-Type : application/pdf

%PDF-1.4

....
%%EOF

34.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

35 Accounts-accountId-invoicesInfo-sorted:GET

36 GET /accounts/{accountId}/invoice?sortField={sortField}

Gets a list of account's invoices sorted by sortField

36.1 Contents

- 1 GET /accounts/{accountId}/invoice?sortField={sortField}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET paginated invoices list call sorted by status
 - ◆ 1.4 See also

36.2 Request

GET /accounts/{accountId}/invoices?sortField={sortField}

36.2.1 Parameters

accountId - *string*
The Id of the account.

sortField - *string*
sortField param. Possible value is 'status'.

sortOrder - *string*
sortOrder param. Possible values are: 'asc','desc'.

36.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

36.3 Response

36.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The accountId was not found.

36.3.2 Response Body

```
{
  "list": [
    {
      "status": "{status}",
      "currency": "{currency}",
      "amount": "{amount}",
      "amountDue": "{amountDue}",
      "description": "{description}",
      "date": "{date}",
      "statusName": "{statusName}",
      "invoiceNumber": "{invoiceNumber}"
    }
  ],
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{account_id}/invoices",
      "rel": "self"
    }
  ]
}
```

36.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about invoices

status - *string*
Status of invoice

currency - *string*
Currency of invoice

ammount - *string*
Amount of invoice

amountDue - *string*
Amount due of invoice

description - *string*
Description of invoice

date - *string*
Date of invoice

statusName - *string*
Status of invoice

invoiceNumber - *string*
Invoice number

links - *list*
[Hypermedia](#) for this resource. Links to self

36.4 Examples

36.4.1 GET paginated invoices list call sorted by status

Request

GET `{{APIBaseURL}}/accounts/test731993/invoices?sortField=status&sortOrder=asc`

Response

HTTP/1.1 200 OK

```
{
  "list" : [
    {
      "status": "invoice_paid",
      "currency": "USD",
      "amount": "1023.7200",
      "amountDue": "0.0000",
      "description": "Invoice #SO-360704",
      "date": "2011-04-19",
      "statusName": "Invoice Paid",
      "invoiceNumber": "SO-360704"
    },
    {
      "status": "invoice_unpaid",
      "currency": "USD",
      "amount": "8692.6500",
      "amountDue": "8262.8600",
      "description": "Invoice #444547",
      "date": "2013-09-23",
      "statusName": "Invoice Unpaid",
      "invoiceNumber": "444547"
    }
  ],
  "links" : [
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?sortField=status&sortOrder=asc",
      "rel" : "self"
    }
  ]
}
```

36.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

37 Accounts-accountId-invoicesInfo:GET

38 GET /accounts/{accountId}/invoices

Retrieves a list of all account's invoices

38.1 Contents

- 1 GET /accounts/{accountId}/invoices
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET paginated invoices list call
 - ◆ 1.4 See also

38.2 Request

GET /accounts/{accountId}/invoices

38.2.1 Parameters

accountId - *string*
The Id of the account.

38.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

38.3 Response

38.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The account Id was not found.

38.3.2 Response Body

```
{
  "list": [
    {
      "status": "{status}",
      "currency": "{currency}",
      "amount": "{amount}",
      "amountDue": "{amountDue}",
      "description": "{description}",
      "date": "{date}",
      "statusName": "{statusName}",
      "invoiceNumber": "{invoiceNumber}"
    }
  ],
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{account_id}/invoices",
      "rel": "self"
    }
  ]
}
```

38.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about invoices

status - *string*

Status of invoice

currency - *string*
Currency of invoice

ammount - *string*
Amount of invoice

amountDue - *string*
Amount due of invoice

description - *string*
Description of invoice

date - *string*
Date of invoice

statusName - *string*
Status of invoice

invoiceNumber - *string*
Invoice number

links - *list*
[Hypermedia](#) for the individual payment method

links - *list*
[Hypermedia](#) for this resource. Links to self

38.4 Examples

38.4.1 GET paginated invoices list call

Request

GET `{{APIBaseURL}}/accounts/test731993/invoices?page=1&pageSize=3`

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "status": "invoice_paid",
    "currency": "USD",
    "amount": "1023.7200",
    "amountDue": "0.0000",
    "description": "Invoice #SO-360704",
    "date": "2011-04-19",
    "statusName": "Invoice Paid",
    "invoiceNumber": "SO-360704"
  } ],
  "links" : [
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=1&pageSize=3",
      "rel" : "self"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=2&pageSize=3",
      "rel" : "next"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/invoices?page=4&pageSize=3",
      "rel" : "last"
    }
  ]
}
```

38.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

39 Accounts-accountId-paymentInfo-default:DELETE

40 DELETE /accounts/{accountId}/paymentInfo/default

Unsets the default payment method. The default payment method is automatically set to "offline" payment. This method is applicable for those franchises where the "offline" payment is allowed.

40.1 Contents

- 1 DELETE
/accounts/{accountId}/paymentInfo/default
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

40.2 Request

DELETE /accounts/{accountId}/paymentInfo/default

40.2.1 Parameters

accountId - *string*
The Id of the account.

40.2.2 URI Parameters

None.

40.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

40.2.4 Request Body

None.

40.3 Response

40.3.1 Status Code

204 No Content
Successful default payment change

404 Not Found
The account Id was not found.

409 Conflict
The "offline" payment method is not allowed

40.3.2 Response Body

None.

40.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

41 Accounts-accountId-paymentInfo-default:PUT

42 PUT /accounts/{accountId}/paymentInfo/default

Sets the default payment method

42.1 Contents

- 1 PUT
/accounts/{accountId}/paymentInfo/default
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

42.2 Request

PUT /accounts/{accountId}/paymentInfo/default

42.2.1 Parameters

accountId - *string*
The Id of the account.

42.2.2 URI Parameters

None.

42.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

42.2.4 Request Body

```
{  
  "paymentInfo": "https://api.hostway.com/accounts/{accountId}/paymentInfo/{paymentId}"  
}
```

42.2.4.1 Parameters

paymentInfo - *string*
The new default payment method. Should contain a link to the payment info that will become the new default.

42.3 Response

42.3.1 Status Code

204 No Content
Successful default payment change

400 Bad Request
Request body was malformed, or an invalid link was submitted for the new default payment.

404 Not Found
The account Id was not found.

42.3.2 Response Body

None.

42.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

43 Accounts-accountId-paymentInfo-paymentInfo-payment:POST

44 POST /accounts/{accountId}/paymentInfo/{paymentInfoId}/payment

Make a one-time payment toward the account balance. First, a "pending" payment transaction will be created, and the payment amount will be reflected in the effective balance but not in the actual balance. After the payment has been processed (asynchronously), the payment status will change to "active" and the payment amount will be reflected in the actual balance as well.

44.1 Contents

- 1 POST
/accounts/{accountId}/paymentInfo/{paymentInfoId}/payment
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Headers
 - ◇ 1.2.3 Response Body
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Make a payment
 - ◆ 1.4 See also

44.2 Request

POST /accounts/{accountId}/paymentInfo/{paymentInfoId}/payment

44.2.1 Request Parameters

accountId
The ID of the account

paymentInfoId
The ID of the payment method

44.2.2 URI Parameters

None

44.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

44.2.4 Request Body

```
{  
  "amount": {paymentAmount}  
}
```

44.2.4.1 Parameters

paymentAmount - *decimal*
The amount of payment in the account's currency. Must be greater than zero.

44.3 Response

44.3.1 Status Code

204 No Content
The payment was successfully submitted

400 Bad Request
The request body was missing a required field or one of the fields did not comply with the requirements (in this case, the response body will contain hints for what went wrong)

401 Unauthorized
Authentication failed

403 Forbidden
Authentication succeeded, but the logged-in user is not allowed to create users in this account. The only users allowed to create users in this account are sales agents and account owners of *this* account. An account owner is any account user with the "sitecontrol_account_owner" role.

404 Not Found
The account or the payment method does not exist

409 Conflict
The payment cannot be submitted due to irregularities with the payment method. For example, this may happen if the credit card is expired, has reached its credit limit or is declined by the payment provider.

44.3.2 Response Headers

None

44.3.3 Response Body

None

44.4 Examples

44.4.1 Make a payment

POST <https://api.hostway.com/accounts/smith123456/paymentInfo/cc-100075384/payment>

```
{  
  "amount": 100.00  
}
```

44.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

45 Accounts-accountId-paymentInfo-paymentInfoId:DELETE

46 DELETE /accounts/{accountId}/paymentInfo/{paymentInfoId}

Removes a given payment method. If the payment method is the current default, it cannot be removed

46.1 Contents

- 1 DELETE /accounts/{accountId}/paymentInfo/{paymentInfoId}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

46.2 Request

DELETE /accounts/{accountId}/paymentInfo/{paymentInfoId}

46.2.1 Parameters

accountId - *string*
The id of the account.

paymentInfoId - *string*
The id of the payment method.
Note: Default payment info can be unset only with agent credentials.

46.2.2 URI Parameters

None.

46.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

46.2.4 Request Body

None.

46.3 Response

46.3.1 Status Code

204 No Content
Successful removal

403 Forbidden
The current user is not allowed to delete this payment method

404 Not Found
The account or payment info was not found

409 Conflict
Cannot delete the current default payment method

46.3.2 Response Body

None.

46.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

47 Accounts-accountId-paymentInfo-paymentInfoId:GET

48 GET /accounts/{accountId}/paymentInfo/{paymentInfoId}

Retrieves an individual active payment method.

48.1 Contents

- 1 GET /accounts/{accountId}/paymentInfo/{paymentInfoId}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

48.2 Request

GET /accounts/{accountId}/paymentInfo/{paymentInfoId}

48.2.1 Parameters

accountId - *string*
The Id of the account.

paymentInfoId - *string*
The Id of the payment method.

48.2.2 URI Parameters

None.

48.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

48.3 Response

48.3.1 Status Code

200 OK
Success

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The account or payment info was not found

48.3.2 Response Body

```
{
  "paymentInfoType": "direct_debit",
  "routingNumber": "564738291",
  "links": [{"href": "https://api.hostway.com/accounts/{accountId}/paymentInfo/dd-100062441", "rel": "self"}],
  "accountNumber": "XXXXXXXXXXXX2332",
  "default": 0
}
```

```
{
  "links": [{"href": "https://api.hostway.com/accounts/{accountId}/paymentInfo/cc-100069155", "rel": "self"}],
  "paymentInfoType": "credit_card",
  "creditCardType": "visa",
  "accountNumber": "XXXXXXXXXXXX1111",
  "expirationDate": "2017-04",
  "default": 1
}
```

48.3.2.1 Parameters

paymentType - *string*
Describes whether the type of the payment method. Valid values are **direct_debit** or **credit_card**

default - *integer*
Whether the payment method is the default. Allowed values are **0** and **1**

accountNumber - *integer*
Available for direct debit and credit card. Contains the masked account number or the masked number of the credit card associated with the payment method

routingNumber - *string*
(*OPTIONAL*)
Only available for direct debit. Contains the routing number of the payment method

creditCardType - *string*
(*OPTIONAL*)
Only available for credit card. Describes the type of the card

expirationDate - *string*
(*OPTIONAL*)
Only available for credit card. Contains the expiration date of the card

links - *list*
[Hypermedia](#) for this resource. Links to self

48.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

49 Accounts-accountId-paymentInfo:GET

50 GET /accounts/{accountId}/paymentInfo

Retrieves a list of all active payment methods.

50.1 Contents

- 1 GET /accounts/{accountId}/paymentInfo
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

50.2 Request

GET /accounts/{accountId}/paymentInfo

50.2.1 Parameters

accountId - *string*
The Id of the account.

50.2.2 URI Parameters

None.

50.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

50.3 Response

50.3.1 Status Code

200 OK
Success

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The account Id was not found.

50.3.2 Response Body

```
{
  "list": [
    {
      "paymentInfoType": "direct_debit",
      "routingNumber": "564738291",
      "links": [
        {
          "href": "https://api.hostway.com/accounts/{accountId}/paymentInfo/dd-100062441",
          "rel": "self"
        }
      ],
      "accountNumber": "XXXXXXXXXXXX2332",
      "default": 0
    },
    {
      "links": [
        {
          "href": "https://api.hostway.com/accounts/{accountId}/paymentInfo/cc-100069155",
          "rel": "self"
        }
      ],
      "paymentInfoType": "credit_card",
      "creditCardType": "visa",
      "accountNumber": "XXXXXXXXXXXX1111",
      "expirationDate": "2017-04",
      "default": 1
    }
  ],
  "default": [
    {
      "href": "https://api.hostway.com/accounts/{accountId}/paymentInfo/default/",
      "rel": "related"
    }
  ],
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{accountId}/paymentInfo/",
      "rel": "self"
    }
  ]
}
```

50.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about specific products

paymentInfoType - *string*
Describes whether the type of the payment method. Valid values are **direct_debit** or **credit_card**

default - *integer*
Whether the payment method is the default. Allowed values are **0** and **1**

accountNumber - *integer*
Available for direct debit and credit card. Contains the masked account number or the masked number of the credit card associated with the payment method

routingNumber - *string*
(*OPTIONAL*)
Only available for direct debit. Contains the routing number of the payment method

creditCardType - *string*
(*OPTIONAL*)
Only available for credit card. Describes the type of the card

expirationDate - *string*
(*OPTIONAL*)
Only available for credit card. Contains the expiration date of the card

links - *list*
[Hypermedia](#) for the individual payment method

links - *list*
[Hypermedia](#) for this resource. Links to self

50.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

51 Accounts-accountId-paymentInfo:POST

52 POST /accounts/{accountId}/paymentInfo

Create a new payment information for a particular account

52.1 Contents

- 1 POST /accounts/{accountId}/paymentInfo
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Headers
 - ◇ 1.2.3 Response Body
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Add an credit card payment info
 - ◇ 1.3.2 Add a direct debit payment info
 - ◆ 1.4 See also

52.2 Request

POST /accounts/{accountId}/paymentInfo

52.2.1 Request Parameters

accountId
The ID of the account

52.2.2 URI Parameters

None

52.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

52.2.4 Request Body

For adding a credit card

```
{
  "paymentInfoType": {paymentInfoType},
  "accountNumber": {accountNumber},
  "cardHolderName": {cardHolderName},
  "expirationDate": {expirationDate},
  "verificationNumber": {verificationNumber},
  "billingAddress": {
    "street1": {street1},
    "street2": {street2},
    "city": {city},
    "stateOrProvince": {stateOrProvince},
    "postalCode": {postalCode},
    "countryCode": {countryCode}
  },
  "routingNumber": {routingNumber},
  "accountOwner": {accountOwner},
  "accountName": {accountName},
  "bankName": {bankName},
  "bankCity": {bankCity},
  "bankCountry": {bankCountry}
}
```

52.2.4.1 Parameters

paymentInfoType - *string*
Required. Type of the payment method. Allowed values are **credit_card** and **direct_debit**.

accountNumber - *string*
Required. Payment account number. This the credit card number or the bank account number.

cardHolderName - *string*
Required.(*For credit cards only*) Name of the card holder. Generally, it is the name that appears on the CC.

expirationDate - *string*
Required.(*For credit cards only*) Expiration date of the credit card. Should be in the **yyyy-m** format.

verificationNumber - *string*
Required.(*For credit cards only*) The CVV number (Card Verification Value) of the credit card.

billingAddress - *dictionary*
Required.(*For credit cards only*) The billing address of the credit card. It's important that the address provided matches the one with which the credit card was registered as the card will be validated through an AVS (Address Verification Service)

- street1 - *string*
Required. The address line 1 of the billing address.
- street2 - *string*
Optional. The address line 2 of the billing address.
- city - *string*
Required. The city of the billing address.
- stateOrProvince - *string*
Optional. The state Or province of the billing address.
- postalCode - *string*
Optional. The postal code of the billing address.
- countryCode - *string*
Optional. The country code of the billing address.

routingNumber - *string*
Required.(*For direct debit only*). The bank routing number.

accountOwner - *string*
Optional.(*For direct debit only*). Name as it Appears on the Check.

accountName - *string*
Optional.(*For direct debit only*). Name for the account.

bankName - *string*
Optional.(*For direct debit only*). The bank name.

bankCity - *string*
Optional.(*For direct debit only*). The bank city.

bankCountry - *string*
Optional.(*For direct debit only*). The bank country.

52.3 Response

52.3.1 Status Code

201 Created
The payment information was successfully created

400 Bad Request
The request body was missing a required field or one of the fields did not comply with the requirements (in this case, the response body will contain hints for what went wrong)

401 Unauthorized
Authentication failed

403 Forbidden
Authentication succeeded, but the logged-in user is not allowed to create users in this account. The only users allowed to create users in this account are sales agents and account owners of *this* account. An account owner is any account user with the "sitecontrol_account_owner" role.

404 Not Found
The account does not exist

409 Conflict
The payment information cannot be created because another payment account with the same information already exists for this user. **OR**
The credit card information has been rejected. Some reasons for CC validation failures are:

- Invalid or expired credit card;
- Invalid CVV number;
- Invalid credit card billing address;
- Black-listed credit card

52.3.2 Response Headers

Location
The URI for the newly created payment information

52.3.3 Response Body

None

52.4 Examples

52.4.1 Add an credit card payment info

POST http://<serverName>/accounts/<accountId>/paymentInfo

```
{
  "paymentInfoType": "credit_card",
  "accountNumber": "1111111111111111",
  "cardHolderName": "John Smith",
  "verificationNumber": "555",
  "creditCardType": "amex",
  "expirationDate": "2015-06",
  "billingAddress": {
    "street1": "Address Line 1",
    "street2": "Address Line 2",
    "countryCode": "US",
    "city": "Chicago",
    "postalCode": "60606",
    "stateOrProvince": "IL"
  }
}
```

Response headers

```
...
Location    http://<serverName>/accounts/<accountId>/paymentInfo/cc-<newly_created_id>
...
```

52.4.2 Add a direct debit payment info

POST http://<serverName>/accounts/<accountId>/paymentInfo

```
{
  "paymentInfoType": "direct_debit",
  "routingNumber": "123456789",
  "accountNumber": "1234",
  "accountName": "My Account Name",
  "accountOwner": "John Smith",
  "bankName": "My Bank",
  "bankCity": "Chicago",
  "bankCountry": "US"
}
```

Response headers

```
...
Location    http://<serverName>/accounts/<accountId>/paymentInfo/dd-<newly_created_id>
...
```

52.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

53 Accounts-accountId-products-productId-action:POST

54 POST /accounts/{accountId}/products/{productId}/action

Upgrades/downgrades existing product.

54.1 Contents

- 1 POST
/accounts/{accountId}/products/{productId}/action
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

54.2 Request

POST /accounts/{accountId}/products/{productId}/action

54.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

54.2.2 URI Parameters

None.

54.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

54.2.4 Request Body

```
{
  "action": "change",
  "actionData": {
    "targetProductSpecification": "https://api.hostway.com/productSpecifications/{productSpecificationId}"
  }
}
```

54.2.4.1 Parameters

action - *string*
The type of the requested action. Currently supported actions:
change - change the product specification (upgrade/downgrade).

actionData - *dictionary*
Contains action-specific data.
change action:
targetProductSpecification - the URI of the new product specification.

54.3 Response

54.3.1 Status Code

204 No Content
The product update was successfully submitted.

400 Bad Request
Request body is invalid.

401 Unauthorized
The authorization credentials were not supplied or are invalid.

404 Not Found
The account or product Ids were not found.

54.3.2 Response Body

None.

54.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

55 Accounts-accountId-products-productId-bandwidth-daily:GET

56 GET

/accounts/{accountId}/products/{productId}/bandwidth/daily?filterType={filterType}&fr

Get a report on the daily bandwidth usage of the product

Each bandwidth usage is filtered by the traffic type(www, ftp), the start date and the end date in ISO format(eg. 2013-02-20)

56.1 Contents

- 1 GET
 - /accounts/{accountId}/products/{productId}/bandwidth/daily?filterType={filterType}&from={filterStartDate}&to={filterEndDate}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET call
 - ◆ 1.4 See also

56.2 Request

GET

/accounts/{accountId}/products/{productId}/bandwidth/daily?filterType={filterType}&from={filterStartDate}&to={filterEndDate}

56.2.1 Request Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

56.2.2 URI Parameters

filterType(optional) - *string - optional*
Should be one of either **www** or **ftp**. In case the total(www + ftp) bandwidth is needed, this parameter can be omitted. If it's omitted, the response will include separate entries for the different types of bandwidth (it will not sum them up for each date).

filterStartDate - *string - required*
Start date of the bandwidth usage report. Should be in **YYYY-MM-DD** format. The usage returned includes usage with dates from **filterStartDate 00:00:00 UTC**. *filterStartDate* must be older or equal to *filterEndDate*.

filterEndDate - *string - required*
Start date of the bandwidth usage report. Should be in **YYYY-MM-DD** format. The bandwidth usage returned includes usage with dates to **filterEndDate 23:59:59 UTC**. *filterEndDate* must be older than today's date in UTC.

showBlanks - *optional*
When the showBlanks parameter is passed, the response body will include values for all dates and types, even if there is no associated traffic.

56.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The **Authentication** credentials of the client application.

56.2.4 Request Body

None

56.3 Response

56.3.1 Status Code

200 OK
Success.

400 Bad Request
A request is considered *bad* any of the following situation is available:

- **filterType** is not one of **www**, **ftp**;
- **filterStartDate** or **filterEndDate** URI parameter is missing;
- **filterEndDate** is older than **filterStartDate**;
- either **filterEndDate** or **filterStartDate** is today's date or a future date;
- **filterEndDate** is more than 100 days after **filterStartDate**

403 Forbidden
Forbidden access to the account.

404 Not Found
The account Id or product Id was not found.

56.3.2 Response Body

```
{
  "list" : [
    {
      "date": {dateISOFormat},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }, ... ,
    {
      "date": {dateISOFormat},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }
  ],
  "links" : [ {
    "href" : "{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/daily?from=2012-01-01"
    "rel" : "self"
  } ]
}
```

56.3.2.1 Parameters

list - *list*

a list of all bandwidth usage records. Each record consists of:

dateISOFormat - *string*

the date for which the bandwidth is collected

trafficType - *string*

type of the traffic(web or ftp)

domainName - *string*

the domain name, for which the bandwidth data is collected

numberOfBytes - *integer*

the bandwidth calculated in bytes

links - *list*

[Hypermedia](#) for this resource. Links to self.

56.4 Examples

56.4.1 GET call

Request

GET

{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/daily?filterType=www&from=2012-01-01

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "date": "2012-01-01",
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }, {
    "date": "2012-01-02",
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 1048576
  }, {
    "date": "2012-01-03",
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 2097152
  } ],
  "links" : [ {
    "href" : "{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/daily?from=2012-01-01"
    "rel" : "self"
  } ]
}
```

56.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

57 Accounts-accountId-products-productId-bandwidth-monthly:GET

58 GET

/accounts/{accountId}/products/{productId}/bandwidth/monthly?filterType={filterType}

Get a report on the monthly bandwidth usage of the product

Each bandwidth usage is filtered by the traffic type(www, ftp), the start date and the end date in ISO format(eg. 2013-02-20)

58.1 Contents

- 1 GET
/accounts/{accountId}/products/{productId}/bandwidth/monthly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET call
 - ◆ 1.4 See also

58.2 Request

GET

/accounts/{accountId}/products/{productId}/bandwidth/monthly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}

58.2.1 Request Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

58.2.2 URI Parameters

filterType(optional) - *string - optional*
Should be one of either **www** or **ftp**. In case the total(www + ftp) bandwidth is needed, this parameter can be omitted. If it's omitted, the response will contain separate entries for each bandwidth type for each month (the values will not be summed for each month).

filterStartDate - *string - required*
Start date of the bandwidth usage report. Should be in **YYYY-MM-DD**(ISO) format. The usage returned includes usage with dates from **filterStartDate 00:00:00 UTC**. *filterStartDate* must be older or equal to *filterEndDate*.

filterEndDate - *string - required*
End date of the bandwidth usage report. Should be in **YYYY-MM-DD**(ISO) format. The bandwidth usage returned includes all dates to **filterEndDate 23:59:59 UTC**. *filterEndDate* must be older than today's date in UTC.

showBlanks - *optional*
When the showBlanks parameter is passed, the response body will include values for all months and types, even if there is no associated traffic.

58.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The **Authentication** credentials of the client application.

58.2.4 Request Body

None

58.3 Response

58.3.1 Status Code

200 OK
Success.

400 Bad Request
A request is considered *bad* any of the following situation is available:

- **filterType** is not one of **www**, **ftp**;
- **filterStartDate** or **filterEndDate** URI parameter is missing;
- **filterEndDate** is older than **filterStartDate**;
- either **filterEndDate** or **filterStartDate** are today's date or a future date;
- **filterEndDate** is more than 100 months after **filterStartDate**

403 Forbidden
Forbidden access to the account.

404 Not Found
The account Id or product Id was not found.

58.3.2 Response Body

```
{
  "list" : [
    {
      "month": {monthNumber},
      "year": {yearNumber},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }, ... ,
    {
      "month": {monthNumber},
      "year": {yearNumber},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }
  ],
  "links" : [ {
    "href" : "{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/monthly?from=2012-01-01",
    "rel" : "self"
  } ]
}
```

58.3.2.1 Parameters

list - *list*

a list of all bandwidth usage records. Each record consists of:

monthNumber - *integer*

the number of the month for which the bandwidth is collected (January = 1, December = 12)

yearNumber - *integer*

the year for which the bandwidth is collected

trafficType - *string*

type of the traffic(www or ftp)

domainName - *string*

the domain name, for which the bandwidth data is collected

numberOfBytes - *integer*

the bandwidth calculated in bytes

links - *list*

[Hypermedia](#) for this resource. Links to self.

58.4 Examples

58.4.1 GET call

Request

```
GET
{{APIBaseURL}}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/monthly?filterType=www&from=2012-01-01
```

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "month": 1,
    "year": 2012,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }, {
    "month": 2,
    "year": 2012,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }, {
    "month": 3,
    "year": 2012,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }, {
    "month": 4,
    "year": 2012,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }, {
    "month": 5,
    "year": 2012,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }
],
  "links" : [ {
    "href": "{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/monthly?filterType=www&from=2012-01-01",
    "rel": "self"
  } ]
}
```

```
},
  "links" : [ {
    "href" : "{{APIBaseURL}}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/monthly?filterType=ww
    "rel" : "self"
  } ]
}
```

58.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

59 Accounts-accountId-products-productId-bandwidth-weekly:GET

60 GET

/accounts/{accountId}/products/{productId}/bandwidth/weekly?filterType={filterType}&

Get a report on the weekly bandwidth usage of the product

Each bandwidth usage is filtered by the traffic type(www, ftp), the start date and the end date in ISO format(eg. 2013-02-20)

60.1 Contents

- 1 GET
/accounts/{accountId}/products/{productId}/bandwidth/weekly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET call
 - ◆ 1.4 See also

60.2 Request

GET

/accounts/{accountId}/products/{productId}/bandwidth/weekly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}

60.2.1 Request Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

60.2.2 URI Parameters

filterType(optional) - *string - optional*
Should be one of either **www** or **ftp**. In case the total(www + ftp) bandwidth is needed, this parameter can be omitted. If it's omitted, the response will include separate entries for each bandwidth type; they will not be summed for each week.

filterStartDate - *string - required*
Start date of the bandwidth usage report. Should be in **YYYY-MM-DD**(ISO) format. The usage returned includes usage with dates from **filterStartDate 00:00:00 UTC**. *filterStartDate* must be older or equal to *filterEndDate*.

filterEndDate - *string - required*
End date of the bandwidth usage report. Should be in **YYYY-MM-DD**(ISO) format. The bandwidth usage returned includes all dates to **filterEndDate 23:59:59 UTC**. *filterEndDate* must be older than today's date in UTC.

showBlanks - *optional*
When the showBlanks parameter is passed, the response body will include values for all weeks and types, even if there is no associated traffic.

60.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The **Authentication** credentials of the client application.

60.2.4 Request Body

None

60.3 Response

60.3.1 Status Code

200 OK
Success.

400 Bad Request
A request is considered *bad* any of the following situation is available:

- **filterType** is not one of **www**, **ftp**;
- **filterStartDate** or **filterEndDate** URI parameter is missing;
- **filterEndDate** is older than **filterStartDate**;
- either **filterEndDate** or **filterStartDate** are today's date or a future date;
- **filterEndDate** is more than 100 weeks after **filterStartDate**

403 Forbidden
Forbidden access to the account.

404 Not Found
The account Id or product Id was not found.

60.3.2 Response Body

```
{
  "list" : [
    {
      "week": {weekNumber},
      "year": {yearNumber},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }, ... ,
    {
      "week": {weekNumber},
      "year": {yearNumber},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }
  ],
  "links" : [ {
    "href" : "{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/weekly?from=2012-01-01",
    "rel" : "self"
  } ]
}
```

60.3.2.1 Parameters

list - *list*

a list of all bandwidth usage records. Each record consists of:

week - *integer*

the week of the year for which the bandwidth is collected. This is a number from 1 to 52 or 53. "Week 1" is whichever week contains the date of January 4 of that year, starting with Monday and ending with Sunday. Any dates within the year that precede this week are considered to be within the last week of the previous year.

year - *integer*

the year for which the bandwidth is collected

trafficType - *string*

type of the traffic(www or ftp)

domainName - *string*

the domain name, for which the bandwidth data is collected

bytes - *integer*

the bandwidth calculated in bytes

links - *list*

[Hypermedia](#) for this resource. Links to self.

60.4 Examples

60.4.1 GET call

Request

GET
{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/weekly?filterType=www&from=2012-01-01

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "week": 23,
    "year": 2013,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }, {
    "week": 24,
    "year": 2013,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 1048576
  }, {
    "week": 25,
    "year": 2013,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 2097152
  } ],
  "links" : [ {
    "href" : "{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/daily?from=2013-06-03",
    "rel" : "self"
  } ]
}
```

60.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

61 Accounts-accountId-products-productId-bandwidth-yearly:GET

62 GET

/accounts/{accountId}/products/{productId}/bandwidth/yearly?filterType={filterType}&f

Get a report on the yearly bandwidth usage of the product

Each bandwidth usage is filtered by the traffic type(www, ftp), the start date and the end date in ISO format(eg. 2013-02-20)

62.1 Contents

- 1 GET
 - /accounts/{accountId}/products/{productId}/bandwidth/yearly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET call
 - ◆ 1.4 See also

62.2 Request

GET

/accounts/{accountId}/products/{productId}/bandwidth/yearly?filterType={filterType}&from={filterStartDate}&to={filterEndDate}

62.2.1 Request Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

62.2.2 URI Parameters

filterType(optional) - *string - optional*
Should be one of either **www** or **ftp**. In case the total(www + ftp) bandwidth is needed, this parameter can be omitted. If it's omitted, the response will contain separate entries for each type for each year; the values will not be summed for each year.

filterStartDate - *string - required*
Start date of the bandwidth usage report. Should be in **YYYY-MM-DD**(ISO) format. The usage returned includes usage with dates from **filterStartDate 00:00:00 UTC**. *filterStartDate* must be older or equal to *filterEndDate*.

filterEndDate - *string - required*
End date of the bandwidth usage report. Should be in **YYYY-MM-DD**(ISO) format. The bandwidth usage returned includes all dates to **filterEndDate 23:59:59 UTC**. *filterEndDate* must be older than today's date in UTC.

showBlanks - *optional*
When the showBlanks parameter is passed, the response body will include values for all years and types, even if there is no associated traffic.

62.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The **Authentication** credentials of the client application.

62.2.4 Request Body

None

62.3 Response

62.3.1 Status Code

200 OK
Success.

400 Bad Request
A request is considered *bad* any of the following situation is available:

- **filterType** is not one of **www**, **ftp**;
- **filterStartDate** or **filterEndDate** URI parameter is missing;
- **filterEndDate** is older than **filterStartDate**;
- either **filterEndDate** or **filterStartDate** are today's date or a future date;
- **filterEndDate** is more than 100 years after **filterStartDate**

403 Forbidden
Forbidden access to the account.

404 Not Found
The account Id or product Id was not found.

62.3.2 Response Body

```
{
  "list" : [
    {
      "year": {yearNumber},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }, ... ,
    {
      "year": {yearNumber},
      "type": {trafficType},
      "domain": {domainName},
      "bytes": {numberOfBytes}
    }
  ],
  "links" : [ {
    "href" : "{{APIBaseURL}}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/yearly?from=2012-01-01",
    "rel" : "self"
  } ]
}
```

62.3.2.1 Parameters

list - *list*

a list of all bandwidth usage records. Each record consists of:

yearNumber - *integer*

the year for which the bandwidth is collected

trafficType - *string*

type of the traffic(www or ftp)

domainName - *string*

the domain name, for which the bandwidth data is collected

numberOfBytes - *integer*

the bandwidth calculated in bytes

links - *list*

[Hypermedia](#) for this resource. Links to self.

62.4 Examples

62.4.1 GET call

Request

GET

{{APIBaseURL}}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/yearly?filterType=www&from=2012-01-01

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "year": 2012,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }, {
    "year": 2013,
    "type": "www",
    "domain": "smithwidgets.com",
    "bytes": 12345678
  }
],
  "links" : [ {
    "href" : "{{APIBaseURL}}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/bandwidth/yearly?filterType=www",
    "rel" : "self"
  } ]
}
```

62.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

63 Accounts-accountId-products-productId-services-serviceId-action:GET

64 POST /accounts/{accountId}/products/{productId}/services/{serviceId}/action

Upgrades/downgrades existing service.

64.1 Contents

- 1 POST
/accounts/{accountId}/products/{productId}/services/{serviceId}/action
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

64.2 Request

POST /accounts/{accountId}/products/{productId}/services/{serviceId}/action

64.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

serviceId - *string*
The Id of the service.

64.2.2 URI Parameters

None.

64.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

64.2.4 Request Body

```
{
  "action": "change",
  "actionData": {
    "targetServiceSpecification": "https://api.hostway.com/serviceSpecifications/{serviceSpecificationId}"
  }
}
```

64.2.4.1 Parameters

action - *string*
The type of the requested action. Currently supported actions:
cost - show the available fees for PTC change
validate - validate if the service specification change is allowed
change - change the service specification (upgrade/downgrade).

actionData - *dictionary*
Contains action-specific data.
validate action:
targetServiceSpecification - the URI of the new service specification.
change action:
targetServiceSpecification - the URI of the new service specification.

64.3 Response

Response for getting the fees

```
{
  "links": [ {
    "href": "http://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId}/action/",
    "rel": "self"
  } ],
  "oneTimeFee": {
    "currency": "USD",
    "amount": "9.95"
  },
  "taxes": {
    "amount": "0"
  },
  "total": {
    "currency": "USD",
    "amount": "19.90"
  },
  "subTotal": {
    "currency": "USD",
    "amount": "19.90"
  }
}
```

```
}
```

Response for the validation

```
{
  "links": [{
    "href": "http://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId}/action/",
    "rel": "self"
  }],
  "allowed": 1
}
```

64.3.1 Status Code

200 No Content

The service update was successfully validated.

204 No Content

The service update was successfully submitted.

400 Bad Request

Request body is invalid.

401 Unauthorized

The authorization credentials were not supplied or are invalid.

404 Not Found

The account, product Id or service Id not found.

64.3.2 Response Body

None.

64.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

65 Accounts-accountId-products-productId-services-serviceId-action:POST

66 POST /accounts/{accountId}/products/{productId}/services/{serviceId}/action

Upgrades/downgrades existing service.

66.1 Contents

- 1 POST
/accounts/{accountId}/products/{productId}/services/{serviceId}/action
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

66.2 Request

POST /accounts/{accountId}/products/{productId}/services/{serviceId}/action

66.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

serviceId - *string*
The Id of the service.

66.2.2 URI Parameters

None.

66.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

66.2.4 Request Body

```
{
  "action": "change",
  "actionData": {
    "targetServiceSpecification": "https://api.hostway.com/serviceSpecifications/{serviceSpecificationId}"
  }
}
```

66.2.4.1 Parameters

action - *string*
The type of the requested action. Currently supported actions:
cost - show the available fees for PTC change
validate - validate if the service specification change is allowed
change - change the service specification (upgrade/downgrade)
changeQuantityCost - get the cost for changing a service quantity
changeQuantity - change the service quantity.

actionData - *dictionary*
Contains action-specific data.
validate action:
targetServiceSpecification - the URI of the new service specification.
change action:
targetServiceSpecification - the URI of the new service specification
carryOverDiscounts - indicates whether current active discounts should be preserved (optional).
changeQuantityCost action:
quantity - the target quantity
carryOverDiscounts - indicates whether current active discounts should be preserved (optional).
changeQuantity action:
quantity - the target quantity
carryOverDiscounts - indicates whether current active discounts should be preserved (optional).

66.3 Response

Response for getting the fees

```
{
  "links": [{
    "href": "http://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId}/action/",
    "rel": "self"
  }],
  "oneTimeFee": {
    "currency": "USD",
    "amount": "9.95"
  },
}
```

```
"taxes": {
  "amount": "0"
},
"total": {
  "currency": "USD",
  "amount": "19.90"
},
"subTotal": {
  "currency": "USD",
  "amount": "19.90"
}
}
```

Response for the validation

```
{
  "links": [{
    "href": "http://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId}/action/",
    "rel": "self"
  }],
  "allowed": 1
}
```

66.3.1 Status Code

200 No Content

The service update was successfully validated.

204 No Content

The service update was successfully submitted.

400 Bad Request

Request body is invalid.

401 Unauthorized

The authorization credentials were not supplied or are invalid.

404 Not Found

The account, product Id or service Id not found.

66.3.2 Response Body

None.

66.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

accounts-accountId-products-productId-services-serviceId-getTargetServiceSpecifications:

68 GET

/accounts/{accountId}/products/{productId}/services/{serviceId}/?getTargetServiceSpe

Get allowed PTC changes

68.1 Contents

- 1 GET
/accounts/{accountId}/products/{productId}/services/{serviceId}/?getTargetServiceSpecifications=1
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

68.2 Request

GET /accounts/{accountId}/products/{productId}/services/{serviceId}/?getTargetServiceSpecifications=1

68.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

serviceId - *string*
The Id of the service.

68.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

68.3 Response

68.3.1 Status Code

200 OK
Success.

401 Unauthorized
Authorization credentials not submitted or invalid.

404 Not Found
The account, product Id or service Id not found.

68.3.2 Response Body

```
{
  "status": "open",
  "name": "Service Name",
  "displayName": "service-related-domain.com",
  "billingCycle": 12,
  "serviceSpecification": "https://api.hostway.com/serviceSpecifications/{serviceSpecificationId}",
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId}",
      "rel": "self"
    },
    {
      "href": "https://api.hostway.com/{detailsLink}",
      "rel": "manage"
    }
  ],
  "changes": [
    {
      "href": "https://api.hostway.com/serviceSpecifications/{serviceSpecificationId}",
      "type": "{type}",
      "name": "Target Service Name",
      "billingCycle": 1
    }
  ]
}
```

68.3.2.1 Parameters

status - *string*
Current status of the service. Allowed values are **new**, **pending**, **open**, **suspended**, **close_pending**, **closed** and **purge_pending**.

name - *string*
Name of the service.

displayName - *string*

Domain name for the service if any exists.

billingCycle - *integer*

Billing cycle of the service in months.

serviceSpecification - *string*

Link to the specification of the current service.

type - *string*

Type of service change. Might be upgrade, downgrade

links - *list*

self - [Hypermedia](#) for the service resource

manage - [Hypermedia](#) for some service types. *detailsLink* format is:

for Domain type: *domains/{domainName}/details*. Please see: [Domains-domainName-details:GET](#)

for Email type: *accounts/{accountId}/email/{domainName}*. Please see: [Accounts-account-email-domain:GET](#)

for Exchange type: *accounts/{accountId}/exchange/{domainName}/mailboxes*. Please see: [GET Exchange Mailboxes list](#)

for Dedicated Server type: *accounts/{accountId}/dedicatedServers/{serverId}*. Please see:

[Accounts-account-dedicatedServers-server:GET](#)

68.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

69 Accounts-accountId-products-productId-services-serviceId:GET

70 GET /accounts/{accountId}/products/{productId}/services/{serviceId}

Get service information.

70.1 Contents

- 1 GET
/accounts/{accountId}/products/{productId}/services/{serviceId}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

70.2 Request

GET /accounts/{accountId}/products/{productId}/services/{serviceId}

70.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

serviceId - *string*
The Id of the service.

70.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

70.3 Response

70.3.1 Status Code

200 OK
Success.

401 Unauthorized
Authorization credentials not submitted or invalid.

404 Not Found
The account, product Id or service Id not found.

70.3.2 Response Body

```
{
  "status": "open",
  "name": "Service Name",
  "displayName": "service-related-domain.com",
  "billingCycle": 12,
  "serviceSpecification": "https://api.hostway.com/serviceSpecifications/{serviceSpecificationId}",
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId}",
      "rel": "self"
    },
    {
      "href": "https://api.hostway.com/{detailsLink}",
      "rel": "manage"
    }
  ]
}
```

70.3.2.1 Parameters

status - *string*
Current status of the service. Allowed values are **new**, **pending**, **open**, **suspended**, **close_pending**, **closed** and **purge_pending**.

name - *string*
Name of the service.

displayName - *string*
Domain name for the service if any exists.

billingCycle - *integer*
Billing cycle of the service in months.

serviceSpecification - *string*
Link to the specification of the current service.

links - *list*
self - *Hypermedia* for the service resource

manage - [Hypermedia](#) for some service types. *detailsLink* format is:
for Domain type: *domains/{domainName}/details*. Please see: [Domains-domainName-details:GET](#)
for Email type: *accounts/{accountId}/email/{domainName}*. Please see: [Accounts-account-email-domain:GET](#)
for Exchange type: *accounts/{accountId}/exchange/{domainName}/mailboxes*. Please see: [GET Exchange Mailboxes list](#)
for Dedicated Server type: *accounts/{accountId}/dedicatedServers/{serverId}*. Please see:
[Accounts-account-dedicatedServers-server:GET](#)

70.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

71 Accounts-accountId-products-productId-services-serviceId:PUT

72 PUT /accounts/{accountId}/products/{productId}/services/{serviceId}'

Updates the status of an existing service.

72.1 Contents

- 1 PUT /accounts/{accountId}/products/{productId}/services/{serviceId}'
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

72.2 Request

PUT /accounts/{accountId}/products/{productId}/services/{serviceId}

72.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

serviceId - *string*
The Id of the service.

72.2.2 URI Parameters

None.

72.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The *Authentication* credentials of the client application.

72.2.4 Request Body

```
{ "status": "new_status" }
```

72.2.4.1 Parameters

status - *string*
Can be one of the following statuses: **open**, **suspended**, **closed**, **purged**.

72.3 Response

72.3.1 Status Code

204 No Content
The status was successfully changed.

400 Bad Request
The request body was invalid.

401 Unauthorized
The authorization credentials were not supplied or are invalid.

404 Not Found
The account or product or service Ids were not found.

409 Conflict
The current status of the product or service does not allow for this change.

72.3.2 Response Body

None.

72.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

73 Accounts-accountId-products-productId-services:GET

74 GET /accounts/{accountId}/products/{productId}/services

Retrieves a list of all services associated with a given product.

74.1 Contents

- 1 GET
/accounts/{accountId}/products/{productId}/services
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

74.2 Request

GET /accounts/{accountId}/products/{productId}/services

74.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

74.2.2 URI Parameters

None.

74.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

74.3 Response

74.3.1 Status Code

200 OK
Success

404 Not Found
The account Id or product Id was not found.

403 Forbidden
Access to **closed** and **suspended** accounts requires elevated permissions

74.3.2 Response Body

```
{
  "list": [
    {
      "name": "Product 1",
      "serviceSpecification": "https://api.hostway.com/serviceSpecifications/{serviceSpecId1}",
      "status": "open",
      "billingCycle": 1,
      "links": [
        { "href": "https://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId1}", "rel": "self" }
      ]
    },
    {
      "name": "Product 2",
      "serviceSpecification": "https://api.hostway.com/serviceSpecifications/{serviceSpecId2}"
      "status": "open",
      "billingCycle": 1,
      "links": [
        { "href": "https://api.hostway.com/accounts/{accountId}/products/{productId}/services/{serviceId2}", "rel": "self" }
      ]
    }
  ],
  "links": [
    { "href": "https://api.hostway.com/accounts/{accountId}/products/{productId}/services", "rel": "self" }
  ]
}
```

74.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about specific services

status - *string*
Describes the status of a service. Valid values are **open**, **closed**, **pending**, or **suspended**

name - *string*
Descriptive name of the service

billingCycle - *integer*

Billing cycle (length of billing period in months) of the service. Allowed values are from **1** to **120**

serviceSpecification - *string*

[Hypermedia](#) for this service resource. Link to the service's specification.

links - *list*

[Hypermedia](#) for this service resource. Links to the service.

links - *list*

[Hypermedia](#) for this resource. Links to self.

74.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

75 Accounts-accountId-products-productId-usage:GET

76 GET

`/accounts/{accountId}/products/{productId}/usage?from={fromDate}&to={toDate}&rate`

Get an aggregated services usage report of a product

Products that currently generate usage data:

- FlexCloud Server
- Exchange 2010 Mailbox Archiving
- Exchange 2010 E-mail Encryption

76.1 Contents

- 1 GET
`/accounts/{accountId}/products/{productId}/usage?from={fromDate}&to={toDate}&rate=true`
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - 1.2.2.2 Usage Types
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET call
 - ◆ 1.4 See also

76.2 Request

GET `/accounts/{accountId}/products/{productId}/usage?from={fromDate}&to={toDate}&rate=true`

76.2.1 Request Parameters

`accountId` - *string*
The Id of the account.

`productId` - *string*
The Id of the product.

76.2.2 URI Parameters

`fromDate` - *string - required*
Start date of the usage report. Should be in **YYYY-MM-DD** format. The usage returned includes usage with dates from **fromDate 00:00:00 UTC**. `fromDate` must be older or equal to `toDate`.

`toDate` - *string - required*
Start date of the usage report. Should be in **YYYY-MM-DD** format. The usage returned includes usage with dates to **toDate 23:59:59 UTC**. `toDate` must be older than today's date in UTC.

`rate` - *string - optional*
If set to `true`, the response will include the price for each usage.

76.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The **Authentication** credentials of the client application.

76.2.4 Request Body

None

76.3 Response

76.3.1 Status Code

200 OK
Success.

400 Bad Request
A request is considered *bad* any of the following situation is available:

- **fromDate** or **toDate** URI parameter is missing;
- **toDate** is older than **fromDate**;
- today is older than or equal to **toDate** or **fromDate**;

403 Forbidden
Forbidden access to regular accounts. Only Sales Agents have access to an *usage* resource.

404 Not Found
The account Id or product Id was not found.

76.3.2 Response Body

```
{  
  "list": [  

```

```

{
  "instance": {instanceName},
  "amount" : {amount},
  "type": {usageType},
  "unit": {usageUnit},
  "ratedAmount": 0.95
}, ...,
{
  "instance": {instanceName},
  "amount": {amount},
  "type": {usageType},
  "unit": {usageUnit},
  "ratedAmount": 10.00
}
],
"links": [
  {
    "href": "{{APIBaseURL}}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/usage?from=2012-01-01&to=2012-01-01",
    "rel": "self"
  }
]
}

```

76.3.2.1 Parameters

`list - list`

a list of all services usage records. Each record consists of:

`instanceName - string`

name of a service instance

`usageType - string`

type of a resource used

`amount - integer`

used resource amount

`unit - string`

measurement unit of the resource used

`ratedAmount - integer/float - (optional)`

the calculated price (charge) for the usage

`links - list`

[Hypermedia](#) for this resource. Links to self.

76.3.2.2 Usage Types

Usage types under monitoring are the following:

Type	Unit	Description
ipaddress	hour	Flexible IP address uptime
loadbalancer	hour	Load Balancer uptime
vm	hour	Virtual Machine uptime
vm.app.office.2007.pro	hour	Virtual Machine uptime for Office 2007 Professional
vm.app.office.2007.std	hour	Virtual Machine uptime for Office 2007 Standard
vm.app.plesk	hour	Virtual Machine uptime for Plesk Control Panel
vm.app.plesk.domain100	hour	Virtual Machine uptime for 100 Plesk domains
vm.app.plesk.powerpack	hour	Virtual Machine uptime for Plesk Powerpack
vm.app.plesk.powerpack	hour	Virtual Machine uptime for Plesk Powerpack
vm.app.sqlserver.std	hour	Virtual Machine uptime for SQL Server Standard Edition
vm.app.sqlserver.web	hour	Virtual Machine uptime for SQL Server Web Edition
vm.app.terminalservices.cal	hour	Virtual Machine uptime for 1 TS CAL
vm.cpu	hour	Virtual Machine CPU uptime
vm.disk.10gb	hour	Virtual Machine uptime for 10GB Disk Space
vm.memory.1gb	hour	Virtual Machine uptime for 1GB of Memory
vm.monitoring.port	hour	Virtual Machine uptime for Port Monitoring
vm.monitoring.url	hour	Virtual Machine uptime for URL Monitoring
vm.os.redhat	hour	Virtual Machine uptime for RedHat OS
vm.traffic.in	mb	Virtual Machine Incoming Traffic
vm.traffic.out	mb	Virtual Machine Outgoing Traffic
vm.traffic.total	mb	Virtual Machine Total Traffic
volume.disk.10gb	hour	Storage Volume uptime for 10GB of Space
vpn	hour	VPN uptime
vpn.remote5	hour	VPN uptime for 5 remote points
vpn.sitetosite	hour	VPN uptime for site-to-site access
hex2013.additionalquota *	mb	Additional quota for HEX 2013

Note: The additional quota unlike the other usage types is not directly retrieved, but rather it is calculated with the following formula:

CoSM = Count of standard mailboxes
QfSM = Quota for standard mailboxes
CoPM = Count of premium mailboxes
QfPM = Quota for premium mailboxes

Additional Quota = Quota - CoSM * QfSM - CoPM * QfPM

76.4 Examples

76.4.1 GET call

Request

GET
{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/usage?from=2012-01-01&to=2012-12-31&rate=tr

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "instance" : "smith182278-Test-VM.1",
    "amount" : 24,
    "type" : "vm.disk.10gb",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.0",
    "amount" : 24,
    "type" : "vm.disk.10gb",
    "unit" : "hour",
    "ratedAmount" : 0.95
  }, {
    "instance" : "smith182278-Test-VM.3",
    "amount" : 24,
    "type" : "vm.disk.10gb",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.2",
    "amount" : 24,
    "type" : "vm.disk.10gb",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.4",
    "amount" : 24,
    "type" : "vm.disk.10gb",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.1",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.0",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.3",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 10.00
  }, {
    "instance" : "smith182278-Test-VM.2",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.5",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 11.05
  }, {
    "instance" : "smith182278-Test-VM.4",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 10.0
  }, {
    "instance" : "smith182278-Test-VM.7",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 10
  }, {
    "instance" : "smith182278-Test-VM.6",
    "amount" : 24,
    "type" : "vm.memory.1gb",
    "unit" : "hour",
    "ratedAmount" : 0.5
  }, {
    "instance" : "",
    "amount" : 6488,
    "type" : "vm.traffic.total",
  }
  ]
}
```

```

    "unit" : "mb",
    "ratedAmount" : 0.99
  }, {
    "instance" : "smith182278-Test-VM",
    "amount" : 354,
    "type" : "vm.traffic.in",
    "unit" : "mb",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM",
    "amount" : 24,
    "type" : "vm.app.plesk",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.1",
    "amount" : 24,
    "type" : "vm.cpu",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.0",
    "amount" : 24,
    "type" : "vm.cpu",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.3",
    "amount" : 24,
    "type" : "vm.cpu",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM.2",
    "amount" : 24,
    "type" : "vm.cpu",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM",
    "amount" : 24,
    "type" : "vm",
    "unit" : "hour",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM",
    "amount" : 6134,
    "type" : "vm.traffic.out",
    "unit" : "mb",
    "ratedAmount" : 0
  }, {
    "instance" : "smith182278-Test-VM",
    "amount" : 1024,
    "type" : "hex2013.additionalquota",
    "unit" : "mb",
    "ratedAmount" : 0
  } ],
  "links" : [ {
    "href" : "{APIBaseURL}/accounts/smith182278/products/smith182278-0000013534cbbc0dc7182cea000aa37f798db0/usage?from=2012-01-01&to=2012-1",
    "rel" : "self"
  } ]
}

```

76.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

77 Accounts-accountId-products-productId:GET

78 GET /accounts/{accountId}/products/{productId}

Get product information and (if requested) possible product changes.

78.1 Contents

- 1 GET /accounts/{accountId}/products/{productId}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

78.2 Request

GET /accounts/{accountId}/products/{productId}

78.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

78.2.2 URI Parameters

getTargetProductSpecifications - *boolean*
Optional. If set to 1 the response will include valid product specification change targets list.

78.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

78.3 Response

78.3.1 Status Code

200 OK
Success.

401 Unauthorized
Authorization credentials not submitted or invalid.

404 Not Found
The account or product Id not found.

78.3.2 Response Body

The response body includes product specification change targets list when the *getTargetProductSpecifications* URI parameter is set to 1.

```
{
  "status": "open",
  "name": "Product Name",
  "displayName": "My custom product name",
  "partnerId": "Partner Id",
  "billingCycle": 12,
  "productSpecification": "https://api.hostway.com/productSpecifications/{productSpecificationId}",
  "changes": [
    {
      "type": "upgrade",
      "id": "{targetProductSpecificationId}",
      "href": "https://api.hostway.com/productSpecifications/{targetProductSpecificationId}"
    }
  ],
  "links": [
    { "href": "https://api.hostway.com/accounts/{accountId}/products/{productId}/", "rel": "self" }
  ]
}
```

78.3.2.1 Parameters

status - *string*
Current status of the product. Allowed values are **new**, **pending**, **open**, **suspended**, **close_pending**, **closed** and **purge_pending**.

name - *string*
Name of the product.

billingCycle - *integer*

Billing cycle of the product in months.

productSpecification - *string*

Link to the specification of the current product.

changes - *list*

A list with the valid product change targets. Returned if *getTargetProductSpecifications=1* query parameter is passed

displayName - *string*

A custom display name for the product.

partnerId - *string*

Partner Id specified for this product.

links - *list*

[Hypermedia](#) for the product resource

78.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

79 Accounts-accountId-products-productId:PUT

80 PUT /accounts/{accountId}/products/{productId}'

Updates the status of an existing product.

80.1 Contents

- 1 PUT /accounts/{accountId}/products/{productId}'
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 See also

80.2 Request

PUT /accounts/{accountId}/products/{productId}

80.2.1 Parameters

accountId - *string*
The Id of the account.

productId - *string*
The Id of the product.

80.2.2 URI Parameters

None.

80.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

80.2.4 Request Body

```
{"status": "new_status"}
```

80.2.4.1 Parameters

status - *string*
Can be one of the following statuses: **open**, **suspended**, **closed**, **purged**.

80.3 Response

80.3.1 Status Code

204 No Content
The status was successfully changed.

400 Bad Request
The request body was invalid.

401 Unauthorized
The authorization credentials were not supplied or are invalid.

404 Not Found
The account or product Ids were not found.

409 Conflict
The current status of the product does not allow for this change.

80.3.2 Response Body

None.

80.4 See also

- [Account Management API](#)
- [Common Features](#)

81 Accounts-accountId-products:GET

82 GET /accounts/{accountId}/products

Retrieves a list of all products associated with a given account.

82.1 Contents

- 1 GET /accounts/{accountId}/products
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Alternative with partner Id
 - ◇ 1.3.1 Parameters
 - ◆ 1.4 See also

82.2 Request

GET /accounts/{accountId}/products

82.2.1 Parameters

accountId - *string*
The Id of the account.

82.2.2 URI Parameters

filterStatus - *string*
Optional. Only products with status matching the value of the parameter will be returned. Allowed values are **pending**, **open**, **suspended**, **closed**.

filterDisplayName - *string*
Optional. Only products with displayName matching the value of the parameter will be returned.

filterPartnerId - *string*
Optional. Only products with partnerId matching the value of the parameter will be returned.

82.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The **Authentication** credentials of the client application.

82.3 Response

82.3.1 Status Code

200 OK
Success

404 Not Found
The account Id was not found.

403 Forbidden
Access to **closed** and **suspended** accounts requires elevated permissions

82.3.2 Response Body

```
{
  "list": [
    {
      "name": "Product 1",
      "displayName": "My custom product name",
      "partnerId": "Partner Id",
      "productSpecification": "https://api.hostway.com/productSpecifications/{productSpecId1}",
      "bandwidth": "https://api.hostway.com/accounts/{accountId}/products/{productId1}/bandwidth/",
      "usage": "https://api.hostway.com/accounts/{accountId}/products/{productId1}/usage/",
      "status": "open",
      "billingCycle": 1,
      "links": [
        { "href": "https://api.hostway.com/accounts/{accountId}/{productId1}", "rel": "self" }
      ]
    },
    {
      "name": "Product 2",
```

```

    "displayName": "My custom product name",
    "partnerId": "Partner Id",
    "productSpecification": "https://api.hostway.com/productSpecifications/{productSpecId2}"
    "bandwidth": "https://api.hostway.com/accounts/{accountId}/products/{productId2}/bandwidth/",
    "usage": "https://api.hostway.com/accounts/{accountId}/products/{productId2}/usage/",
    "status": "open",
    "billingCycle": 1,
    "links": [
      {"href": "https://api.hostway.com/accounts/{accountId}/products/{productId2}", "rel": "self"}
    ],
  },
  "links": [
    {"href": "https://api.hostway.com/accounts/{accountId}/products", "rel": "self"}
  ]
}

```

82.3.2.1 Parameters

list - *list*

A collection of dictionaries containing information about specific products

status - *string*

Describes the status of a product. Valid values are **open**, **closed** or **suspended**

name - *string*

Descriptive name of the product

partnerId - *string*

Partner Id specified for this product.

billingCycle - *integer*

Billing cycle (length of billing period in months) of the service. Allowed values are from **1** to **120**

productSpecification - *string*

[Hypermedia](#) for this product resource. Link to the product's specification.

bandwidth - *string*

[Hypermedia](#) for this product resource. Link to the bandwidth's information.

usage - *string*

[Hypermedia](#) for this product resource. Link to the usage's information.

links - *list*

[Hypermedia](#) for this product resource. Links to the product.

links - *list*

[Hypermedia](#) for this resource. Links to self.

82.4 Alternative with partner Id

As an alternative to using the account Id the client can use a partner-specific Id set via the account attribute `partner_account_id`. The response to this request is the same as the response to a request using the account Id.

GET /partnerAccounts/{partnerAccountId}/products

82.4.1 Parameters

partnerAccountId - *string*

Partner account Id set via the `partner_account_id` attribute.

82.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

83 Accounts-accountId-transactionsInfo-filtered-type:GET

84 GET /accounts/{accountId}/transactions?filterType={filterType}

Gets a list of account's transactions filtered by transaction type

84.1 Contents

- 1 GET
/accounts/{accountId}/transactions?filterType={filterType}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET paginated transaction list call
 - ◆ 1.4 See also

84.2 Request

GET /accounts/{accountId}/transactions?filterType={filterType}

84.2.1 Parameters

accountId - *string*
The Id of the account.

filterType - *string*
Type of transaction. Possible values are: 'charge', 'charge_back', 'payment', 'fee', 'refund', 'credit', 'cc_chargeback', 'charge_adjustment', 'discount', 'credit_adjustment', 'credit_proration', 'charge_proration', 'payment_adjustment', 'tax', 'cc_chargeback_reversal', 'usage_charge', 'write_off', 'undo_write_off', 'usage_discount'

84.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

84.3 Response

84.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The accountId was not found.

84.3.2 Response Body

```
{
  "list": [
    {
      "startDate": "{startDate}",
      "endDate": "{endDate}",
      "amount": {amount},
      "description": "{description}",
      "currency": "{currency}",
      "date": "{date}",
      "productSpecification": "",
      "serviceNumber": "{serviceNumber}",
      "type": "{type}",
      "number": {number}
    }
  ],
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{account_id}/transactions",
      "rel": "self"
    }
  ]
}
```

84.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about transactions

startDate - *string*
Start date of transaction

endDate - *string*
End date of transaction

ammount - *integer*
Amount of transaction

description - *string*
Description of transaction

currency - *string*
Currency symbol of transaction

date - *string*
Date of transaction

productSpecification - *link*
URL link to the product specification

serviceNumber - *string*
Service number info

type - *string*
Type of transaction. Possible values are: 'charge', 'charge_back', 'payment', 'fee', 'refund', 'credit', 'cc_chargeback', 'charge_adjustment', 'discount', 'credit_adjustment', 'credit_proration', 'charge_proration', 'payment_adjustment', 'tax', 'cc_chargeback_reversal', 'usage_charge', 'write_off', 'undo_write_off', 'usage_discount'

number - *int*
Transaction's number

links - *list*
[Hypermedia](#) for the individual payment method

links - *list*
[Hypermedia](#) for this resource. Links to self

84.4 Examples

84.4.1 GET paginated transaction list call

Request

GET `{{APIBaseURL}}/accounts/test731993/transactions?filterType=payment&page=1&pageSize=10`

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "startDate": "2012-04-23",
    "amount": 0.87,
    "endDate": "2012-05-18",
    "description": "Recurring fee Exchange 2010 Archiving, core.email.exchange2010.archiving 04/23/2012 through 05/18/2012",
    "currency": "USD",
    "date": "2012-04-23",
    "productSpecification": "{{APIBaseURL}}/productSpecifications/test731993-00000137a32daee9e2b08715000a30fe33d32c",
    "serviceNumber": "test731993-00000136dedf26dfd9ec22fd000alddba01dal",
    "type": "payment",
    "number": 101393524
  } ],
  "links" : [
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/transactions?page=1&pageSize=10",
      "rel" : "self"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/transactions?page=2&pageSize=10",
      "rel" : "next"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/transactions?page=44&pageSize=10",
      "rel" : "last"
    }
  ]
}
```

84.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

85 Accounts-accountId-transactionsInfo-sorted:GET

86 GET

/accounts/{accountId}/transactions?sortField={sortField}&sortOrder={sortOrder}

Gets a list of account's transactions sorted by sortField param

86.1 Contents

- 1 GET
/accounts/{accountId}/transactions?sortField={sortField}&sortOrder={sortOrder}
- ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
- ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
- ◆ 1.3 Examples
 - ◇ 1.3.1 GET sorted by amount value transaction list call
- ◆ 1.4 See also

86.2 Request

GET /accounts/{accountId}/transactions?sortField={sortField}&sortOrder={sortOrder}

86.2.1 Parameters

accountId - *string*
The Id of the account.

sortField - *string*
sortField param. Possible values are: 'amount','type'.

sortOrder - *string*
sortOrder param. Possible values are: 'asc','desc'.

86.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

86.3 Response

86.3.1 Status Code

200 OK
Success

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
The user lacks the necessary permissions to access the resource

404 Not Found
The accountId was not found.

86.3.2 Response Body

```
{
  "list": [
    {
      "startDate": "{startDate}",
      "endDate": "{endDate}",
      "amount": {amount},
      "description": "{description}",
      "currency": "{currency}",
      "date": "{date}",
      "productSpecification": "",
      "serviceNumber": "{serviceNumber}",
      "type": "{type}",
      "number": {number}
    }
  ],
  "links": [
    {
      "href": "https://api.hostway.com/accounts/{account_id}/transactions",
      "rel": "self"
    }
  ]
}
```

86.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about transactions

startDate - *string*
Start date of transaction

endDate - *string*
End date of transaction

amount - *integer*
 Amount of transaction

description - *string*
 Description of transaction

currency - *string*
 Currency symbol of transaction

date - *string*
 Date of transaction

productSpecification - *link*
 URL link to the product specification

serviceNumber - *string*
 Service number info

type - *string*
 Type of transaction. Possible values are: 'charge', 'charge_back', 'payment', 'fee', 'refund', 'credit', 'cc_chargeback', 'charge_adjustment', 'discount', 'credit_adjustment', 'credit_proration', 'charge_proration', 'payment_adjustment', 'tax', 'cc_chargeback_reversal', 'usage_charge', 'write_off', 'undo_write_off', 'usage_discount'

number - *int*
 Transaction's number

links - *list*
[Hypermedia](#) for the individual payment method

links - *list*
[Hypermedia](#) for this resource. Links to self

86.4 Examples

86.4.1 GET sorted by amount value transaction list call

Request

GET `{{APIBaseURL}}/accounts/test731993/transactions?sortField=amount&sortOrder=asc`

Response

HTTP/1.1 200 OK

```

{
  "list" : [
    {
      "startDate": "2012-04-23",
      "amount": 0.87,
      "endDate": "2012-05-18",
      "description": "Recurring fee Exchange 2010 Archiving, core.email.exchange2010.archiving 04/23/2012 through 05/18/2012",
      "currency": "USD",
      "date": "2012-04-23",
      "productSpecification": "{{APIBaseURL}}/productSpecifications/test731993-00000137a32dae9e2b08715000a30fe33d32c",
      "serviceNumber": "test731993-00000136dedf26dfd9ec22fd000a1ddba01da1",
      "type": "payment",
      "number": 101393524
    },
    {
      "startDate": "2013-05-04",
      "amount": 1.89,
      "endDate": "2013-06-18",
      "description": "Payment - Notes: - Reference: VUGTI5F6NL0IRJSA17OR CC# 1111 (ProcessSOPayment)",
      "currency": "USD",
      "date": "2013-05-04",
      "productSpecification": "{{APIBaseURL}}/productSpecifications/test731993-00000737a32dae9e2b08715000a30fe13d32f",
      "serviceNumber": "test731993-00000136dedf26dfd9ec22fd000a1ddba02da2",
      "type": "payment",
      "number": 101395701
    }
  ],
  "links" : [
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/transactions?sortField=amount&sortOrder=asc",
      "rel" : "self"
    }
  ]
}

```

86.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

87 Accounts-accountId-transactionsInfo:GET

88 GET /accounts/{accountId}/transactions

Retrieves a list of all account's transactions

88.1 Contents

- 1 GET /accounts/{accountId}/transactions
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 GET paginated transaction list call
 - ◆ 1.4 See also

88.2 Request

GET /accounts/{accountId}/transactions

88.2.1 Parameters

accountId - *string*
The Id of the account.

88.2.2 URI Parameters

None.

88.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.
Accept - *HTTP Accept header* [2]
Possible values are *application/json*, *text/csv*. If omitted json will be returned.

88.3 Response

88.3.1 Status Code

- 200 OK
Success
- 401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.
- 403 Forbidden
The user lacks the necessary permissions to access the resource
- 404 Not Found
The account Id was not found.

88.3.2 Response Body

```
{
  {
    "list": [{
      "startDate": "{startDate}",
      "endDate": "{endDate}",
      "amount": {amount},
      "description": "{description}",
      "currency": "{currency}",
      "date": "{date}",
      "productSpecification": "",
      "serviceNumber": "{serviceNumber}",
      "type": "{type}",
      "number": {number}
    }],
    "links" : [{
      "href" : "https://api.hostway.com/accounts/{account_id}/transactions",
      "rel" : "self"
    }
  ]
}
```

88.3.2.1 Parameters

list - *list*

A collection of dictionaries containing information about transactions

startDate - *string*

Start date of transaction

endDate - *string*

End date of transaction

ammount - *integer*

Amount of transaction

description - *string*

Description of transaction

currency - *string*

Currency symbol of transaction

date - *string*

Date of transaction

productSpecification - *link*

URL link to the product specification

serviceNumber - *string*

Service number info

type - *string*

Type of transaction. Possible values are: 'charge', 'charge_back', 'payment', 'fee', 'refund', 'credit', 'cc_chargeback', 'charge_adjustment', 'discount', 'credit_adjustment', 'credit_proration', 'charge_proration', 'payment_adjustment', 'tax', 'cc_chargeback_reversal', 'usage_charge', 'write_off', 'undo_write_off', 'usage_discount'

number - *int*

Transaction's number

links - *list*

[Hypermedia](#) for the individual payment method

links - *list*

[Hypermedia](#) for this resource. Links to self

88.4 Examples

88.4.1 GET paginated transaction list call

Request

GET `{{APIBaseURL}}/accounts/test731993/transactions?page=1&pageSize=10`

Response

HTTP/1.1 200 OK

```
{
  "list" : [ {
    "startDate": "2012-04-23",
    "amount": 0.87,
    "endDate": "2012-05-18",
    "description": "Recurring fee Exchange 2010 Archiving, core.email.exchange2010.archiving 04/23/2012 through 05/18/2012",
    "currency": "USD",
    "date": "2012-04-23",
    "productSpecification": "{{APIBaseURL}}/productSpecifications/test731993-00000137a32dae9e2b08715000a30fe33d32c",
    "serviceNumber": "test731993-00000136dedf26dfd9ec22fd000alddba01dal",
    "type": "charge",
    "number": 101393524
  } ],
  "links" : [
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/transactions?page=1&pageSize=10",
      "rel" : "self"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/transactions?page=2&pageSize=10",
      "rel" : "next"
    },
    {
      "href" : "{{APIBaseURL}}/accounts/test731993/transactions?page=44&pageSize=10",
      "rel" : "last"
    }
  ]
}
```

88.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

89 Accounts-accountId-user:PUT

90 PUT /accounts/{accountId}/user

Updates account user credentials.

90.1 Contents

- 1 PUT /accounts/{accountId}/user
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◆ 1.3 Examples
 - ◆ 1.4 See also

90.2 Request

PUT /accounts/{accountId}/user

90.2.1 Parameters

accountId - *string*
The Id of the account.

90.2.2 URI Parameters

None.

90.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

Content-Type
application/json

90.2.4 Request Body

```
{
  "userName": "{username}",
  "password": "{password}",
  "securityQuestion": "{securityQuestionCode}",
  "securityAnswer": "{securityAnswer}"
}
```

90.2.4.1 Parameters

userName
Required. A name for the user that's about to be created. This needs to be unique within the entire security realm (not just within the account).

password
Required only if securityQuestion and securityAnswer were not provided. A password for the user. This needs to be between 8 and 50 characters long inclusive, must contain at least one alpha *and* at least one numeric character, and must not contain any of the following characters: & ` ' " \ / < > \$

securityQuestionCode
Required if password was not provided. The reference code for a question that the user can use to recover the password in case (s)he forgot it. This must be selected from a pre-approved list of questions. If securityQuestion was provided, a securityAnswer must also be provided.

securityAnswer
Required if password was not provided. An answer to the security question. If the user provides the correct answer during password recovery, the password will be reset and (s)he will be allowed to access the account. If securityAnswer was provided, a securityQuestion must also be provided. The API will automatically normalize the security answer by removing starting and ending spaces, lowercase the security answer and replace each sequence of whitespaces with just one single space character.

90.3 Response

90.3.1 Status Code

204 No Content
The user was successfully updated

400 Bad Request
The request body was missing a required field or one of the fields did not comply with the requirements (in this case, the response body will contain hints for what went wrong)

401 Unauthorized
Authentication failed

403 Forbidden

Authentication succeeded, but the logged-in user is not allowed to update users in this account. The only users allowed to update this user resource are *this very user*, account owner(s) of *this* account, and sales/support agents. An account owner is any account user with the "sitecontrol_account_owner" role.

404 Not Found

Either the account or user does not exist

415 Unsupported Media Type

The Content-Type header was not specified or the specified type is not supported. Use application/json.

90.4 Examples

Request

```
PUT /accounts/test-account/user
{ "userName":"testuser", "password":"password12" }
```

Response

401 Unauthorized

Request

```
PUT /accounts/test-account/user
Authorization: Basic base64_encode(username:password)
{ "userName":"testuser", "password":"password12" }
```

Response

```
415 Unsupported Media Type
{"badMediaType": {"message": "Unsupported media", "code": 415, "details": ""}}
```

Request

```
PUT /accounts/test-account/user
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{ "userName":"notvalid-user", "password":"password12" }
```

Response

```
404 Not Found
{"itemNotFound": {"message": "Resource not found", "code": 404, "details": ""}}
```

Request

```
PUT /accounts/test-account/user
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{ "userName":"testuser", "password":"" }
```

Response

```
400 Bad Request
{"badRequest": {"message": "POST data error", "code": 400, "details": {"password": "Required"}}
```

Request

```
PUT /accounts/test-account/user
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{ "userName":"testuser", "password":"pass" }
```

Response

```
400 Bad Request
{"computeFault": {"message": "The password should be at least 8 characters", "code": 400, "details": ""}}
```

Request

```
PUT /accounts/test-account/user
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{ "userName":"testuser", "password":"pa$$word" }
```

Response

400 Bad Request

```
{"computeFault": {"message": "The password must contains digits and letters", "code": 400, "details": ""}}
```

Request

```
PUT /accounts/test-account/user  
Authorization: Basic base64_encode(username:password)  
Content-Type: application/json
```

```
{ "userName": "testuser", "password": "password" }
```

Response

400 Bad Request

```
{"computeFault": {"message": "The password must contains at least one digit", "code": 400, "details": ""}}
```

Request

```
PUT /accounts/test-account/user  
Authorization: Basic base64_encode(username:password)  
Content-Type: application/json
```

```
{ "userName": "testuser", "password": "password12" }
```

Response

204 No Content

90.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

91 Accounts-accountId-users-username-roles:GET

92 GET /accounts/{accountId}/users/{userName}/roles

Retrieve the roles of a user

92.1 Contents

- 1 GET
/accounts/{accountId}/users/{userName}/roles
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

92.2 Request

GET /accounts/{accountId}/users/{userName}/roles

92.2.1 Request Parameters

accountId - *string*
The Id of the account
userName - *string*
The username of the user

92.2.2 URI Parameters

None

92.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The [Authentication](#) credentials of the client application.

92.3 Response

92.3.1 Status Code

200 OK
Success

304 Not changed
The user's roles haven't changed since the last request; the client should use its locally cached copy

401 Unauthorized
Authentication failed

403 Forbidden
Authentication succeeded, but the authenticated user is not allowed to view this user's roles. The only authenticated users allowed to view these roles are *this very user*, the account owner(s) of *this* account, and sales/support agents. An account owner is a user within the account that has the role "sitecontrol_account_owner".

404 Not found
Either the account or the requested user does not exist.

92.3.2 Response Body

```
{
  "list": [
    "role1",
    "role2"
  ],
  "links": [
    { "href": "https://api.hostway.com/accounts/{accountId}/users/{userName}/roles", "rel": "self" }
  ]
}
```

92.3.2.1 Parameters

accountId
The ID of the account
userName
The username of this user

92.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

93 Accounts-accountId-users-username-roles:PUT

94 PUT /accounts/{accountId}/users/{userName}/roles

Modify the roles of a user.

94.1 Contents

- 1 PUT
/accounts/{accountId}/users/{userName}/roles
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Allowed Roles
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◆ 1.3 See also

94.2 Request

PUT /accounts/{accountId}/users/{userName}/roles

94.2.1 Request Parameters

accountId
The ID of the account
userName
The username of the user

94.2.2 URI Parameters

None

94.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

94.2.4 Request Body

List of new roles to be set to the requested user

```
[  
  "role1",  
  "role2"  
]
```

94.2.4.1 Allowed Roles

Can be retrieved with GET /roles call.

94.3 Response

94.3.1 Status Code

204 No Content
The user's roles were successfully updated
400 Bad Request
The request body was malformed, or the specified roles are not allowed, or the account owner was attempting to demote himself/herself which is not allowed. (in this case, the response body will contain hints for what went wrong)
401 Unauthorized
Authentication failed
403 Forbidden
Authentication succeeded, but the logged-in user is not allowed to update the roles of this user. The only users allowed to update the roles of this user are *this very user*, account owner(s) of *this* account, and sales/support agents. An account owner is any account user with the "sitecontrol_account_owner" role. An account owner may not remove their own "sitecontrol_account_owner" role, though they may do this to other account owners.
404 Not Found
Either the account or user does not exist

94.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

95 Accounts-accountId-users-userName:DELETE

96 DELETE /accounts/{accountId}/users/{userName}

Delete a user

96.1 Contents

- 1 DELETE /accounts/{accountId}/users/{userName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Headers
 - ◇ 1.2.3 Response Body
 - ◆ 1.3 See also

96.2 Request

DELETE /accounts/{accountId}/users/{userName}

96.2.1 Request Parameters

accountId - *string*
The Id of the account
userName - *string*
The username of the user to delete

96.2.2 URI Parameters

None

96.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

96.3 Response

96.3.1 Status Code

204 No Content
User was successfully deleted
401 Unauthorized
Authentication failed
403 Forbidden
Authentication succeeded, but the authenticated user is not allowed to delete this user resource. The only authenticated users allowed to delete this user resource are the account owner(s) of *this* account, and sales/support agents. Account owners cannot delete themselves. An account owner is a user within the account that has the role "sitecontrol_account_owner".
404 Not found
Either the account or the requested user does not exist.

96.3.2 Response Headers

None

96.3.3 Response Body

None

96.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

97 Accounts-accountId-users-userName:GET

98 GET /accounts/{accountId}/users/{userName}

Return the details of one specific user

98.1 Contents

- 1 GET
/accounts/{accountId}/users/{userName}
- ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
- ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
- ◆ 1.3 See also

98.2 Request

GET /accounts/{accountId}/users/{userName}

98.2.1 Request Parameters

accountId - *string*
The Id of the account

userName - *string*
The username of the user to retrieve

98.2.2 URI Parameters

None

98.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

98.3 Response

98.3.1 Status Code

200 OK
Success

304 Not changed
The user details haven't changed since the last request; the client should use its locally cached copy

401 Unauthorized
Authentication failed

403 Forbidden
Authentication succeeded, but the authenticated user is not allowed to view this user resource. The only authenticated users allowed to view this user resource are *this very user*, the account owner(s) of *this* account, and sales/support agents, the user which authorization credentials match the requested user's credentials. An account owner is a user within the account that has the role "sitecontrol_account_owner".

404 Not found
Either the account or the requested user does not exist.

98.3.2 Response Body

```
{
  "userName": "{userName}",
  "securityQuestion": "{securityQuestionCode}",
  "roles": "https://api.hostway.com/accounts/{accountId}/users/{userName}/roles",
  "links": [
    { "href": "https://api.hostway.com/accounts/{accountId}/users/{userName}", "rel": "self" }
  ]
}
```

98.3.2.1 Parameters

accountId
The ID of the account

userName
The username of this user

securityQuestionCode
The security question code of this user (this will be used to refer to a list of security questions)

98.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

99 Accounts-accountId-users-userName:PUT

100 PUT /accounts/{accountId}/users/{userName}

Modify the password and/or security question/answer of a user

100.1 Contents

- 1 PUT
/accounts/{accountId}/users/{userName}
- ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 = Request Body
 - 1.1.4.1 Parameters
- ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Headers
 - ◇ 1.2.3 Response Body
- ◆ 1.3 See also

100.2 Request

PUT /accounts/{accountId}/users/{userName}

100.2.1 Request Parameters

accountId
The ID of the account

userName
The username of the user

100.2.2 URI Parameters

None

100.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

100.2.4 = Request Body

```
{  
  "userName": "{userName}",  
  "password": "{password}",  
  "securityQuestion": "{securityQuestionCode}",  
  "securityAnswer": "{securityAnswer}"  
}
```

100.2.4.1 Parameters

userName
Required. A name for the user that's about to be created. This needs to be unique within the entire security realm (not just within the account).

password
Required only if securityQuestion and securityAnswer were not provided. A password for the user. This needs to be between 8 and 50 characters long inclusive, must contain at least one alpha *and* at least one numeric character, and must not contain any of the following characters: & ` ' " \ / < > \$

securityQuestionCode
Required if password was not provided. The reference code for a question that the user can use to recover the password in case (s)he forgot it. This must be selected from a pre-approved list of questions. If securityQuestion was provided, a securityAnswer must also be provided.

securityAnswer
Required if password was not provided. An answer to the security question. If the user provides the correct answer during password recovery, the password will be reset and (s)he will be allowed to access the account. If securityAnswer was provided, a securityQuestion must also be provided. The API will automatically normalize the security answer by removing starting and ending spaces, lowercase the security answer and replace each sequence of whitespaces with just one single space character.

100.3 Response

100.3.1 Status Code

204 No Content
The user was successfully updated

400 Bad Request
The request body was missing a required field or one of the fields did not comply with the requirements (in this case, the response body will contain hints for what went wrong)

401 Unauthorized
Authentication failed

403 Forbidden

Authentication succeeded, but the authenticated user is not allowed to view this user resource. The only authenticated users allowed to view this user resource are *this very user*, the account owner(s) of *this* account, and sales/support agents, the user which authorization credentials match the requested user's credentials. An account owner is a user within the account that has the role "sitecontrol_account_owner".

404 Not Found

Either the account or user does not exist

100.3.2 Response Headers

None

100.3.3 Response Body

None

100.4 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

101 Accounts-accountId-users:GET

102 GET /accounts/{accountId}/users

Returns a list of users in an account

102.1 Contents

- 1 GET /accounts/{accountId}/users
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See Also

102.2 Request

GET /accounts/{accountId}/users

102.2.1 Request Parameters

accountId - *string*
The Id of the account

102.2.2 URI Parameters

None

102.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

102.3 Response

102.3.1 Status Code

- 200 OK
Success
- 304 Not changed
The list of users hasn't changed since the last request; the client should use its locally cached copy instead.
- 401 Unauthorized
The authorization header contained an invalid value
- 403 Forbidden
Successfully authenticated, but client is not allowed to view the list of users. Access is only permitted for agents and account owners of this account. An account owner is a user with the "sitecontrol_account_owner" role.
- 404 Not Found
The account in question does not exist

102.3.2 Response Body

```
{
  "list": [
    {
      "userName": "{userName}",
      "details": "https://api.hostway.com/accounts/{accountId}/users/{userName}"
    },
    ...
  ],
  "links": [
    { "href": "https://api.hostway.com/accounts/{accountId}/users", "rel": "self" }
  ]
}
```

102.3.2.1 Parameters

accountNumber
the Id of the account

userName
the username of the user

102.4 See Also

- [Account](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

103 Accounts-accountId-users:POST

104 POST /accounts/{accountId}/users

Create a new user under an account

104.1 Contents

- 1 POST /accounts/{accountId}/users
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Headers
 - ◇ 1.2.3 Response Body
 - ◆ 1.3 Allowed Security Questions
 - ◆ 1.4 See also

104.2 Request

POST /accounts/{accountId}/users

104.2.1 Request Parameters

accountId
The ID of the account

104.2.2 URI Parameters

None

104.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

104.2.4 Request Body

```
{
  "userName": "{userName}",
  "password": "{password}",
  "securityQuestion": "{securityQuestionCode}",
  "securityAnswer": "{securityAnswer}"
}
```

104.2.4.1 Parameters

userName
Required. A name for the user that's about to be created. This needs to be unique within the entire security realm (not just within the account).

password
Required. A password for the user. This needs to be between 8 and 50 characters long inclusive, must contain at least one alpha *and* at least one numeric character, and must not contain any of the following characters: & ` ' " \ / < > \$

securityQuestionCode
Required. The reference code for a question that the user can use to recover the password in case (s)he forgot it. This must be selected from a pre-approved list of questions.

securityAnswer
Required. An answer to the security question. If the user provides the correct answer during password recovery, the password will be reset and (s)he will be allowed to access the account. The API will automatically normalize the security answer by removing starting and ending spaces, lowercase the security answer and replace each sequence of whitespaces with just one single space character.

104.3 Response

104.3.1 Status Code

201 Created

- The user was successfully created
- 400 Bad Request
The request body was missing a required field or one of the fields did not comply with the requirements (in this case, the response body will contain hints for what went wrong)
- 401 Unauthorized
Authentication failed
- 403 Forbidden
Authentication succeeded, but the logged-in user is not allowed to create users in this account. The only users allowed to create users in this account are sales agents and account owners of *this* account. An account owner is any account user with the "sitecontrol_account_owner" role.
- 404 Not Found
The account does not exist
- 409 Conflict
The user cannot be created because another user exists with the same username.

104.3.2 Response Headers

Location
The URI for the newly created user

104.3.3 Response Body

None

104.4 Allowed Security Questions

TBD

104.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

105 Accounts-accountId:DELETE

106 DELETE /accounts/{accountId}

Purges a user account. This call requires elevated permissions.

106.1 Contents

- 1 DELETE /accounts/{accountId}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Alternative with partner Id
 - ◇ 1.3.1 Parameters
 - ◆ 1.4 See also

106.2 Request

DELETE /accounts/{accountId}

106.2.1 Parameters

accountId - *string*
The Id of the account.

106.2.2 URI Parameters

None.

106.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

106.3 Response

106.3.1 Status Code

204 No Content
Success.

404 Not Found
The account Id was not found.

106.3.2 Response Body

None.

106.4 Alternative with partner Id

As an alternative to using the account Id the client can use a partner-specific Id set via the account attribute `partner_account_id`. The response to this request is the same as the response to a request using the account Id.

DELETE /partnerAccounts/{partnerAccountId}

106.4.1 Parameters

partnerAccountId - *string*
Partner account Id set via the `partner_account_id` attribute.

106.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

107 Accounts-accountId:GET

108 GET /accounts/{accountId}

Get details about an account.

108.1 Contents

- 1 GET /accounts/{accountId}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Conditional Hypermedia
 - ◆ 1.4 Alternative with partner Id
 - ◇ 1.4.1 Parameters
 - ◆ 1.5 See also

108.2 Request

GET /accounts/{accountId}

108.2.1 Parameters

accountId - *string*
The Id of the account.

108.2.2 URI Parameters

None.

108.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

108.3 Response

108.3.1 Status Code

200 OK
Success.

403 Forbidden
Forbidden access to **closed** and **suspended** accounts by a user not authorized for such access.

404 Not Found
The account Id was not found.

108.3.2 Response Body

```
{
  "accountNumber": "{accountId}",
  "createdDate": "{createdDate}",
  "currency": "{currency}",
  "status": "open",
  "paymentInfo": [
    {"href": "https://api.hostway.com/accounts/{account-number}/paymentInfo", "rel": "related"}
  ],
  "products": [
    {"href": "https://api.hostway.com/accounts/{account-number}/products", "rel": "related"}
  ],
  "exchange": [
    {"href": "https://api.hostway.com/accounts/{account-number}/exchange", "rel": "related"}
  ],
  "dedicatedServers": [
    {"href": "https://api.hostway.com/accounts/{account-number}/dedicatedServers", "rel": "related"}
  ],
}
```

```

"contacts": [
  {"href": "https://api.hostway.com/accounts/{account-number}/contacts", "rel": "related"}
],
"email": [
  {"href": "https://api.hostway.com/accounts/{account-number}/email", "rel": "related"}
],
"domains": [
  {"href": "https://api.hostway.com/accounts/{account-number}/domains", "rel": "related"}
],
"attributes": [
  {"href": "https://api.hostway.com/accounts/{account-number}/attributes", "rel": "related"}
],
"flexcloud": [
  {"href": "https://api.hostway.com/accounts/{account-number}/flexcloud", "rel": "related"}
],
"users": [
  {"href": "https://api.hostway.com/accounts/{account-number}/users", "rel": "related"}
],
"links": [
  {"href": "https://api.hostway.com/accounts/{account-number}/", "rel": "self"}
]
}

```

108.3.2.1 Parameters

accountNumber - *string*

The Id of the account

createdDate - *date*

The date account was created in format Y-m-d

currency - *string*

An ISO-4217[2] currency code assigned to the account.

status - *string*

status of the account. Valid values are **open**, **closed** or **suspended**. Accounts with certain status may not be accessible by regular users.

paymentInfo - *list*

(OPTIONAL)

Hypermedia to a resource of the payment instruments associated with the account.

users - *list*

(OPTIONAL)

Hypermedia to a resource of the users associated with the account.

products - *list*

(OPTIONAL)

Hypermedia to a resource containing all the products of the account. This value will appear depending on the user's access rights.

dedicatedServers - *list*

(OPTIONAL)

Hypermedia to a resource containing all the dedicated server products of the account.

exchange - *list*

(OPTIONAL)

Hypermedia to a resource containing all the Exchange domains of the account.

contacts - *list*

(OPTIONAL)

Hypermedia to a resource handling the contact details of the account.

email - *list*

(OPTIONAL)

Hypermedia to a resource containing all the email domains of the account.

domains - *list*

(OPTIONAL)

Hypermedia linking to the domain name management of the account.

attributes - *list*

(OPTIONAL)

Hypermedia to a partner attributes of the account.

flexcloud - *list*

(OPTIONAL)

Hypermedia to the Flexcloud management of the account.

links - *list*

Hypermedia for this resource. Links to self.

108.4 Conditional Hypermedia

Hypermedia fields marked as optional will appear on the basis of the accessing user's permissions.

108.5 Alternative with partner Id

As an alternative to using the account Id the client can use a partner-specific Id set via the account attribute `partner_account_id`. The response to this request is the same as the response to a request using the account Id.

GET /partnerAccounts/{partnerAccountId}

108.5.1 Parameters

`partnerAccountId` - *string*

Partner account Id set via the `partner_account_id` attribute.

108.6 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

109 Accounts-accountId:PUT

110 PUT /accounts/{accountId}

Changes the status of an account. This operation requires elevated permissions.

110.1 Contents

- 1 PUT /accounts/{accountId}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Alternative with partner Id
 - ◇ 1.3.1 Parameters
 - ◆ 1.4 See also

110.2 Request

PUT /accounts/{accountId}

110.2.1 Parameters

accountId - *string*
The Id of the account.

110.2.2 URI Parameters

None.

110.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

110.2.4 Request Body

```
{  
  "status": "suspended"  
}
```

110.2.4.1 Parameters

status - *string*
The new status of the account. Accepted values are **open**, **closed** or **suspended**.

110.3 Response

110.3.1 Status Code

204 No Content
Successful account status change

404 Not Found
The account Id was not found.

110.3.2 Response Body

None.

110.4 Alternative with partner Id

As an alternative to using the account Id the client can use a partner-specific Id set via the account attribute `partner_account_id`. The response to this request is the same as the response to a request using the account Id.

```
PUT /partnerAccounts/{partnerAccountId}
```

110.4.1 Parameters

`partnerAccountId` - *string*

Partner account Id set via the `partner_account_id` attribute.

110.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

111 Accounts:GET

112 GET /accounts

Retrieves a list of all accounts authorized for access with the provided credentials.

112.1 Contents

- 1 GET /accounts
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Listing current account only when provided credentials don't give elevated access
 - ◇ 1.3.2 Paginated request when more than one account is returned
 - ◆ 1.4 See also

112.2 Request

GET /accounts

112.2.1 URI Parameters

filterStatus - *string*

Optional. Only accounts with status matching the value of the parameter will be returned. Allowed values are **pending**, **open**, **suspended**, **closed**.

page - *integer*

Optional. Specifies which page should be displayed. Requires a pageSize parameter to also be provided.

pageSize - *integer*

Optional. Specifies the number of entries to be displayed on a page. Default size is 100. Requires a page parameter to also be provided.

112.2.2 Request Headers

Authorization - *HTTP Authorization header [1]*

The **Authentication** credentials of the client application.

112.3 Response

112.3.1 Status Code

200 OK
Success

400 Bad Request

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

112.3.2 Response Body

```
{
  "list" : [{
    "status" : "{status}",
    "links" : [{
      "href" : "https://api.hostway.com/accounts/{account-number}/",
      "rel" : "self"
    }],
    "accountNumber" : "{account-number}",
    "createdDate" : "{created-date}",
    "currency": "{currency}"
  }],
  "links" : [{
    "href" : "https://api.hostway.com/accounts?pageSize={pageSize}&page={page}",
    "rel" : "self"
  },
  {
    "href" : "https://api.hostway.com/accounts?pageSize={pageSize}&page={page}",
    "rel" : "next"
  },
  {
    "href" : "https://api.hostway.com/accounts?pageSize={pageSize}&page={page}",
    "rel" : "last"
  }
]
```

112.3.2.1 Parameters

list - *list*

A collection of dictionaries containing information about specific account

status - *string*

Describes the status of the account. Valid values are **pending**, **open**, **suspended**, **closed**

accountNumber - *string*

The Id of the account

createdDate - *date*

The date account was created in format Y-m-d

currency - *string*

An ISO-4217[2] currency code assigned to the account.

links - *list*

[Hypermedia](#) for this account resource. Links to the account.

links - *list*

[Hypermedia](#) Link to self and other resources relevant to the account lists

112.4 Examples

112.4.1 Listing current account only when provided credentials don't give elevated access

Request

GET /accounts

Response

```
{
  "list": [{
    "status": "open",
    "links": [{
      "href": "https://api.hostway.com/accounts/test-account/",
      "rel": "self"
    }],
    "accountNumber": "test-account",
    "createdDate": "2009-12-07",
    "currency": "USD"
  }],
  "links": [{
    "href": "https://api.hostway.com/accounts",
    "rel": "self"
  }]
}
```

112.4.2 Paginated request when more than one account is returned

Request

GET /accounts?pageSize=2&page=1

Response

```
{
  "list" : [{
    "status" : "open",
    "links" : [{
      "href" : "https://api.hostway.com/accounts/test-account-1/",
      "rel" : "self"
    }],
    "accountNumber" : "test-account-1",
    "createdDate": "2009-12-07",
    "currency": "USD"
  },
  "status" : "closed",
  "links" : [{
    "href" : "https://api.hostway.com/accounts/test-account-2/",
    "rel" : "self"
  }],
  "accountNumber" : "test-account-2",
  "createdDate": "20012-11-01",
  "currency": "CAD"
}],
  "links" : [{
    "href" : "https://api.hostway.com/accounts?pageSize=2&page=1",
    "rel" : "self"
  },
  {
    "href" : "https://api.hostway.com?pageSize=2&page=2",
    "rel" : "next"
  },
  {
    "href" : "https://api.hostway.com?pageSize=2&page=11637",
    "rel" : "last"
  }
]
```

112.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

113 Avas-SSO:GET

114 GET /accounts/{accountId}/avas/{domain}/SSOLink

Returns a SSO Link for the antispam control panel.

114.1 Contents

- 1 GET /accounts/{accountId}/avas/{domain}/SSOLink
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Successful Request

114.2 Request

GET /accounts/{accountId}/avas/{domain}/SSOLink

114.2.1 Request Parameters

accountId - *string*
The unique account Id.

domain - *string*
The domain that needs to be managed.

114.2.2 URI Parameters

None.

114.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

114.3 Response

114.3.1 Status Code

200 OK
Success

401 Unauthorized
The request was sent without an *Authorization* header, or the user represented by the value of this header was not properly authenticated.

404 Not Found
The specified account or domain does not exist or is purged.

409 Conflict
The API was unable to generate a token

114.3.2 Response Body

```
{
  "ssoLink" : "{ssoLink}",
  "links" : [ {
    "href" : "https://api.hostway.com/accounts/{accountId}/avas/{domain}/SSOLink",
    "rel" : "self"
  } ]
}
```

114.3.2.1 Parameters

ssoLink - *string*
Contains a link for SSO login into the antispam control panel.

links - *array*
Hypermedia for this resource.

114.4 Examples

114.4.1 Successful Request

Request

GET /accounts/acc1234/avas/example.com/SSOLink
Authorization: Basic {Base64-encoded username & password string}

Response

200 OK

```
{
  "ssoLink" : "https://demo1.spambrand.com?authticket=1234567890abcdefghij1234567890abcdefghij",
  "links" : [ {
    "href" : "{APIBaseURL}/accounts/acc1234/avas/example.com/SSOLink",
    "rel" : "self"
  } ]
}
```

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

115 Email Validation:POST

POST /validations/email

Validate a registered email.

115.1 Contents

- 1 Request
 - ◆ 1.1 Parameters
 - ◆ 1.2 URI Parameters
 - ◆ 1.3 Request Headers
 - ◆ 1.4 Request Body
 - ◇ 1.4.1 Parameters
- 2 Response
 - ◆ 2.1 Status Code
- 3 Example
 - ◆ 3.1 Success scenario
 - ◆ 3.2 Invalid authentication token scenario
- 4 See also

115.2 Request

POST /validations/email/

115.2.1 Parameters

None.

115.2.2 URI Parameters

None.

115.2.3 Request Headers

Content-Type
application/json

115.2.4 Request Body

```
{
  "authString": "{authentication_token}",
  "v": "{initialization_vector}"
}
```

115.2.4.1 Parameters

authString - *string*
Authentication token

v - *string*
Initialization vector

115.3 Response

115.3.1 Status Code

204 No Content
Success.

400 Bad Request
One of the required attributes is missing from the request body.

409 Conflict
The supplied authentication token is invalid or expired.

115.4 Example

115.4.1 Success scenario

Request

POST /validations/email
Content-Type: application/json

```
{"authString": "{authentication_token}", "v": "{authentication_vector}"}
```

Response

204 No Content

115.4.2 Invalid authentication token scenario

Request

```
POST /validations/email
Content-Type: application/json

{"authString": "{invalid_authentication_token}", "v": "{authentication_vector}"}
```

Response

```
409 Conflict
```

115.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

116 Initiate-Email-Verification-Action:POST

117 POST /accounts/{account_number}/contacts/action

Apply an action on contacts email e.g. **initiate email verification**

117.1 Contents

- 1 POST /accounts/{account_number}/contacts/action
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Expected Response Codes
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Initiate email verification request
 - ◆ 1.4 See also

117.2 Request

POST /accounts/{account_number}/contacts/action

117.2.1 Request Parameters

account_number - *string*
The user account that owning the email.

117.2.2 Request Headers

Authorization - *HTTP Authorization header* [1]
The **Authentication** credentials of the client application.

117.2.3 Request Body

```
{
  "action": "verifyEmail",
  "actionData": "{contactinfo_type}"
}
```

action - *string*
The action that should be performed. Allowed values are:
verifyEmail - email verification.

actionData - *string*
Data to be send with the action. This must be a contact information type.
contactinfo_type - There are 4 types of contact information that can be used with the request: **regular**, **billing**, **administrator** or **technical**.

117.3 Response

117.3.1 Expected Response Codes

202 Accepted
Success

400 Bad Request
The supplied action is invalid.

401 Unauthorized
The supplied credentials are invalid.

403 Forbidden
The authorized user does not have permissions to access the resource. The only user(s) who has access to the contacts email must be a sales agent, the account owner, or the account technical admin.

404 Not Found
The contacts email for the provided contact info type does not exist.

409 Conflict
The contacts email is already verified.

117.4 Examples

117.4.1 Initiate email verification request

Request

```
POST /accounts/acct515393/contacts/action
Content-Type:application/json; Charset=UTF-8
Authorization: Basic {Base64 string}
```

```
{
  "action": "verifyEmail",
```

```
} "actionData": "regular"
```

Response

202 Accepted

117.5 See also

- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

118 PartnerAccounts-partnerAccountId-attributes-attribute:DELETE

119 DELETE /partnerAccounts/{partnerAccountId}/attributes/{attribute}

Updates attribute value.

119.1 Contents

- 1 DELETE /partnerAccounts/{partnerAccountId}/attributes/{attribute}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Failure updating not existing attribute
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

119.2 Request

DELETE /partnerAccounts/{partnerAccountId}/attributes/{attribute}

119.2.1 Parameters

partnerAccountId - *string*
The Id of the partner account.

119.2.2 URI Parameters

attribute - *string*

119.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

Content-Type
application/json

119.3 Response

119.3.1 Status Code

204 No Content
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The partner account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

119.4 Examples

119.4.1 Sending unauthorized request

Request

```
DELETE /partnerAccounts/partner-account-id/attributes/test_attribute
```

Response

```
401 Unauthorized
```

119.4.2 Sending request without setting Content-Type

Request

```
DELETE /partnerAccounts/partner-account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
```

Response

```
415 Unsupported Media Type
```

119.4.3 Failure updating not existing attribute

Request

```
DELETE /partnerAccounts/partner-account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

404 Not Found

119.4.4 Success scenario

Request

```
DELETE /partnerAccounts/partner-account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

204 No Content

119.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

120 PartnerAccounts-partnerAccountId-attributes-attribute:GET

121 GET /partnerAccounts/{partnerAccountId}/attributes/{attribute}

Get attribute data.

121.1 Contents

- 1 GET /partnerAccounts/{partnerAccountId}/attributes/{attribute}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Requesting not existing attribute
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

121.2 Request

GET /partnerAccounts/{partnerAccountId}/attributes/{attribute}

121.2.1 Parameters

partnerAccountId - *string*
The Id of the partner account.

121.2.2 URI Parameters

attribute - *string*

121.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

Content-Type
application/json

121.3 Response

121.3.1 Status Code

200 OK
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The partner account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

121.3.2 Response Body

```
{
  "name": "{name}",
  "value": {value},
  "links": [
    {
      "location": "{APIBaseURL}/partnerAccounts/{partnerAccountId}/attributes/{attribute}/",
      "rel": "self"
    }
  ]
}
```

121.3.2.1 Parameters

name - *string*
Descriptive name of the attribute

value - *string*
Value of the attribute

links - *list*

[Hypermedia](#) for this resource. Links to self.

121.4 Examples

121.4.1 Sending unauthorized request

Request

```
GET /partnerAccounts/partner-account-id/attributes/test_attribute
```

Response

```
401 Unauthorized
```

121.4.2 Sending request without setting Content-Type

Request

```
GET /partnerAccounts/partner-account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
```

Response

```
415 Unsupported Media Type
```

121.4.3 Requesting not existing attribute

Request

```
GET /partnerAccounts/partner-account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
404 Not Found
```

121.4.4 Success scenario

Request

```
GET /partnerAccounts/partner-account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
200 OK
```

```
{
  "name": "test_attribute",
  "value": "123",
  "links": [{"href": "{APIBaseURL}/partnerAccounts/partner-account-id/attributes/test_attribute/", "rel": "related"}]
}
```

121.5 See also

- [Account Attribute Management](#)
- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

122 PartnerAccounts-partnerAccountId-attributes-attribute:PUT

123 PUT /partnerAccounts/{partnerAccountId}/attributes/{attribute}

Updates attribute value.

123.1 Contents

- 1 PUT
/partnerAccounts/{partnerAccountId}/attributes/{attribute}
- ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
- ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
- ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Failure updating not existing attribute
 - ◇ 1.3.4 Failure updating attribute with missing required value
 - ◇ 1.3.5 Success scenario
- ◆ 1.4 See also

123.2 Request

```
PUT /partnerAccounts/{partnerAccountId}/attributes/{attribute}
```

123.2.1 Parameters

partnerAccountId - *string*
The Id of the partner account.

123.2.2 URI Parameters

attribute - *string*

123.2.3 Request Headers

Authorization - *HTTP Authorization header [1]*
The [Authentication](#) credentials of the client application.

Content-Type
application/json

123.2.4 Request Body

```
{  
  "value": "{value}"  
}
```

123.2.4.1 Parameters

value - *string*

123.3 Response

123.3.1 Status Code

204 No Content
Success.

400 Bad Request
The format of the request body is invalid.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The partner account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

123.4 Examples

123.4.1 Sending unauthorized request

Request

```
PUT /partnerAccounts/partner-account-id/attributes/test_attribute
{"value": "test"}
```

Response

401 Unauthorized

123.4.2 Sending request without setting Content-Type

Request

```
PUT /partnerAccounts/partner-account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
{"value": "test"}
```

Response

415 Unsupported Media Type

123.4.3 Failure updating not existing attribute

Request

```
PUT /partnerAccounts/partner-account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"value": "test"}
```

Response

404 Not Found

123.4.4 Failure updating attribute with missing required value

Request

```
PUT /partnerAccounts/partner-account-id/attributes/not_existing_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{}
```

Response

400 Bad Request
{"value": "Required"}

123.4.5 Success scenario

Request

```
PUT /partnerAccounts/partner-account-id/attributes/test_attribute
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"value": "test"}
```

Response

204 No Content

123.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

124 PartnerAccounts-partnerAccountId-attributes:GET

125 GET /partnerAccounts/{partnerAccountId}/attributes

Get attributes list.

125.1 Contents

- 1 GET /partnerAccounts/{partnerAccountId}/attributes
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Getting list of attributes of not existing partner
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

125.2 Request

GET /partnerAccounts/{partnerAccountId}/attributes

125.2.1 Parameters

partnerAccountId - *string*
The Id of the partner account.

125.2.2 URI Parameters

None.

125.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The [Authentication](#) credentials of the client application.

Content-Type
application/json

125.3 Response

125.3.1 Status Code

200 OK
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The partner account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

125.3.2 Response Body

```
{
  "list": [
    {
      "name": "{name}",
      "value": "{value}",
      "links": [{"href": "https://api.hostway.com/partnerAccounts/{partnerAccountId}/attributes/{name}", "rel": "related"}]
    }
  ],
  "links": [
    {"href": "https://api.hostway.com/partnerAccounts/{partnerAccountId}/attributes", "rel": "self"}
  ]
}
```

125.3.2.1 Parameters

list - *list*
A collection of dictionaries containing information about specific attribute
name - *string*

Descriptive name of the attribute

value - *string*
Value of the attribute

NOTE: The "partner_account_id" attribute will not be returned in this list.

links - *list*
[Hypermedia](#) for this product resource. Links to the attribute.

links - *list*
[Hypermedia](#) for this resource. Links to self.

125.4 Examples

125.4.1 Sending unauthorized request

Request

```
GET /partnerAccounts/partner-account-id/attributes
```

Response

```
401 Unauthorized
```

125.4.2 Sending request without setting Content-Type

Request

```
GET /partnerAccounts/partner-account-id/attributes
Authorization: Basic base64_encode(username:password)
```

Response

```
415 Unsupported Media Type
```

125.4.3 Getting list of attributes of not existing partner

Request

```
GET /partnerAccounts/not-existing-partner-account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
404 Not Found
```

125.4.4 Success scenario

Request

```
GET /partnerAccounts/partner-account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
```

Response

```
200 OK
{
  "list": [
    {
      "name": "test_attribute",
      "value": "123",
      "links": [{"href": "{{APIBaseURL}}/partnerAccounts/partner-account-id/attributes/test_attribute/", "rel": "related"}]
    }
  ],
  "links": [
    {
      "href": "{{APIBaseURL}}/partnerAccounts/partner-account-id/attributes/",
      "rel": "self"
    }
  ]
}
```

125.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

126 PartnerAccounts-partnerAccountId-attributes:POST

127 POST /partnerAccounts/{partnerAccountId}/attributes

Get attributes list.

127.1 Contents

- 1 POST
/partnerAccounts/{partnerAccountId}/attributes
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Sending unauthorized request
 - ◇ 1.3.2 Sending request without setting Content-Type
 - ◇ 1.3.3 Failure creating attribute for not existing partner
 - ◇ 1.3.4 Success scenario
 - ◆ 1.4 See also

127.2 Request

POST /partnerAccounts/{partnerAccountId}/attributes

127.2.1 Parameters

partnerAccountId - *string*
The Id of the partner account.

127.2.2 URI Parameters

None.

127.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

Content-Type
application/json

127.2.4 Request Body

```
{  
  "name": "test_attribute",  
  "value": "value"  
}
```

127.2.4.1 Parameters

name - *string*
Descriptive name of the attribute

value - *dictionary*
Value of the attribute

127.3 Response

127.3.1 Status Code

201 Created
Success.

401 Unauthorized
The supplied credentials are invalid or do not provide permissions for this operation.

403 Forbidden
Forbidden access to a user not authorized for such access.

404 Not Found
The partner account Id or attribute were not found.

415 Unsupported Media Type
The Content-Type header was not specified or the specified type is not supported. Use application/json.

127.4 Examples

127.4.1 Sending unauthorized request

Request

```
POST /partnerAccounts/partner-account-id/attributes
{"name": "test_Account", "value": "test"}
```

Response

```
401 Unauthorized
```

127.4.2 Sending request without setting Content-Type

Request

```
POST /partnerAccounts/partner-account-id/attributes
Authorization: Basic base64_encode(username:password)
{"name": "test_Account", "value": "test"}
```

Response

```
415 Unsupported Media Type
```

127.4.3 Failure creating attribute for not existing partner

Request

```
POST /partnerAccounts/not-existing-partner-account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"name": "test_Account", "value": "test"}
```

Response

```
404 Not Found
```

127.4.4 Success scenario

Request

```
POST /partnerAccounts/partner-account-id/attributes
Authorization: Basic base64_encode(username:password)
Content-Type: application/json
{"name": "test_Account", "value": "test"}
```

Response

```
201 Created
```

127.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

128 ServiceAttributes-core.cloud.hscc

129 Cloud Compute Service Attributes

Attributes used when configuring the Cloud computing service:

129.1 Contents

- 1 Cloud Compute Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

129.2 Attributes Body

```
{  
  "accountNumber": "acct123456",  
}
```

129.2.1 Parameters

accountNumber
The account Number of the Hostway account.

129.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

130 ServiceAttributes-core.dedserver

131 Dedicated Server Service Attributes

Attributes used when configuring the Dedicated Server service.

131.1 Contents

- 1 Dedicated Server Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

131.2 Attributes Body

```
{  
  "serverName": "MyServer",  
  "serverLocation": "Tampa"  
}
```

131.2.1 Parameters

serverName

A name for the dedicated server.

serverLocation

The name of the data center where the server is to be located.

131.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

132 ServiceAttributes-core.dnr

133 Domain Name Registration Service Attributes

Attributes used when configuring the Domain Name Registration service.

133.1 Contents

- 1 Domain Name Registration Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

133.2 Attributes Body

```
{
  "domainName": "mydomain.com",
  "operation": "register",
  "term": 1,
  "authInfo": "mysecret",
  "nameservers": [ "ns1.myprovider.com", "ns2.myprovider.com" ],
  "extraParams": [{"name": "us_nexus", "value": "C11"}, {"name": "us_purpose", "value": "P1"}],
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "firstName": "John",
      "lastName": "Doe",
      "company": "ACME Inc.",
      "street1": "123 First Str.",
      "city": "Chicago",
      "zipOrPostal": "60000",
      "stateOrProvince": "IL",
      "countryCode": "US",
      "email1": "john.doe@test.com",
      "phone1": "+1.123.5555555" },
    "administrative": {
      "type": "administrative",
      ...
    },
    "billing": {
      "type": "billing",
      ...
    },
    "technical": {
      "type": "technical",
      ...
    }
  }
}
```

133.2.1 Parameters

- domainName - *string*
The domain name
- operation - *string*
"register" to register a domain
"transfer" to transfer a domain
- term - *integer*
the registration term (period) in years
- authInfo - *string*
authentication info, required for transfer operation
- nameservers - *list of strings*
domain name servers (DNS) for the domain
- extraParams - *list of extra params* (optional)
List of additional parameters required for some TLDs. Each parameter is an object with "name" and "value" fields. For example, .us registration requires the fields "us_nexus" which can be one of the following [C11, C12, C21, C31, C32] - see [What are Domain Name Application Purpose Codes?](#) and "us_purpose" which can be one of [P1, P2, P3, P4, P5] - see [What do the Nexus Codes mean?](#) You can find more information on extra parameters in the [GET /domains/tlds/{tld}/extrparams](#) documentation.
- company - *string* (optional)
The company or organization name
- firstName - *string*
The first name of the contact person
- lastName - *string*
The last name of the contact person
- street1 - *string*
The street address
- street2 - *string* (optional)
The second line of the street address (if any)
- city - *string*
The city
- countryCode - *string*
The international 2-letter country code
- stateOrProvince - *string*
The state or province. For US states use the 2-letter state code
- zipOrPostal - *string*
The postal code
- email1 - *string*
The e-mail address of the contact person
- phone1 - *string*
The phone number of the contact person in valid [E.164](#) format, e.g. +1.3209270625

133.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

134 ServiceAttributes-core.fax.number

135 Fax Over Email Number Attributes

Attributes used for the FOE fax number.

135.1 Contents

- 1 Fax Over Email Number Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

135.2 Attributes Body

```
{
  "areaCode": "{area_code}",
  "name": "{name}"
}
```

135.2.1 Parameters

`areaCode` - *string*
(*REQUIRED*)
Standard 3 digit US [area code](#) (e.g. *312*, *412*).
Represents one of the supported area codes for a fax number within a particular ZIP area.
Please refer to the [FOE Supported Area Codes by ZIP code API](#) page for a detailed information about the supported area codes.

`name` - *string*
Customer friendly name/label for the fax number.

135.3 See also

- [Supported Area Codes by ZIP code](#)
- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

136 ServiceAttributes-core.fax.user

137 Fax Over Email User Attributes

Attributes used for the FOE user.

137.1 Contents

- 1 Fax Over Email User Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

137.2 Attributes Body

```
{  
  "zipCode": "{zip_code}",  
}
```

137.2.1 Parameters

zipCode - *string*
(*REQUIRED*)
US standard ZIP code - both ZIP code 5 and ZIP code 5 plus 4 formats are supported (e.g. 98001, 98001-9000).
Represents the zip code of the user.

137.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

138 ServiceAttributes-core.online backup.account

139 Online Backup Account Attributes

Attributes used for the online backup account.

139.1 Contents

- 1 Online Backup Account Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

139.2 Attributes Body

```
{  
  "defaultUserQuota": {default_user_quota},  
}
```

139.2.1 Parameters

defaultUserQuota - *integer*
(*REQUIRED*)

The default quota in MB for each user to be created.

139.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

140 ServiceAttributes-core.sharepoint

141 Sharepoint Service Attributes

Attributes used when configuring the Web Hosting service.

141.1 Contents

- 1 Sharepoint Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

141.2 Attributes Body

```
{  
  "domain": "mydomain.com", "is_shared": 0|1, "name": "portalname", "template_key": "templatename"  
}
```

141.2.1 Parameters

domain The domain name of the web hosting site.
is_shared
name
template_key

141.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

142 ServiceAttributes-core.sharepoint2007

143 SharePoint 2007 Attributes

Attributes used when configuring the SharePoint 2007.

143.1 Contents

- 1 SharePoint 2007 Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

143.2 Attributes Body

```
{
  "domain": "{domain}",
  "is_shared": 0|1,
  "name": "{name}",
  "template_key": "{template_key}"
}
```

143.2.1 Parameters

domain - *String*

(REQUIRED)

The domain name for the SharePoint product.

is_shared - *Integer*

(REQUIRED)

Flag that specifies whether the domain attribute is shared domain or not. Possible values are **1** or **0** (a.k.a true or false).

name - *String*

(REQUIRED)

Sub domain name which will be the prefix of the actual **FQDN** for the SharePoint portal resource.

template_key - *String*

(OPTIONAL)

Specific template text key for the SharePoint resource.

143.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

144 ServiceAttributes-core.spamexperts

145 AVAS Service Attributes

Attributes used when configuring the Antivirus Antispam service.

145.1 Contents

- [1 AVAS Service Attributes](#)
 - ◆ [1.1 Attributes Body](#)
 - ◇ [1.1.1 Parameters](#)
 - ◆ [1.2 See also](#)

145.2 Attributes Body

```
{  
  "domain": "mydomain.com"  
}
```

145.2.1 Parameters

`domain` The email domain to be filtered.

145.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

146 ServiceAttributes-core.subdomain

147 Sub-Domain Registration Service Attributes

Attributes used when configuring the Sub-Domain Registration service.

147.1 Contents

- [1 Sub-Domain Registration Service Attributes](#)
 - ◆ [1.1 Attributes Body](#)
 - ◇ [1.1.1 Parameters](#)
 - ◆ [1.2 See also](#)

147.2 Attributes Body

```
{
  "name": "sub-domain",
  "auto_generate": "0"
}
```

147.2.1 Parameters

name The name of the sub-domain.

auto_generate When set to *1* forces the sub-domain name to be generated. In this case if the *name* is specified, it will be ignored.

147.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

148 ServiceAttributes-core.vps.virtuozzo.server

149 VPS Virtuozzo Server Attributes

Attributes used when configuring the Virtuozzo server.

149.1 Contents

- [1 VPS Virtuozzo Server Attributes](#)
 - ◆ [1.1 Attributes Body](#)
 - ◇ [1.1.1 Parameters](#)
 - ◆ [1.2 See also](#)

149.2 Attributes Body

```
{  
  "serverHostName": "host",  
}
```

149.2.1 Parameters

serverHostName - *String*
(*OPTIONAL*)
The hostname of the new server.

149.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

150 ServiceAttributes-core.webhosting

151 Web Hosting Service Attributes

Attributes used when configuring the Web Hosting service.

151.1 Contents

- 1 Web Hosting Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

151.2 Attributes Body

```
{  
  "domain": "mydomain.com"  
}
```

151.2.1 Parameters

domain The domain name of the web hosting site.

151.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

152 ServiceAttributes-core.websplanet

153 Websplanet Attributes

Attributes used when configuring the Websplanet service.

153.1 Contents

- 1 Websplanet Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

153.2 Attributes Body

```
{  
  "domain": "ng-test-drago-140416a.com",  
  "email": "test@example.com"  
}
```

153.2.1 Parameters

domain Domain name for Websplanet.

email email address.

153.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

154 ServiceAttributes-dedicated.server.firewall

155 Dedicated Server Firewall Service Attributes

Attributes used when configuring the Dedicated Server Firewall service.

155.1 Contents

- 1 Dedicated Server Firewall Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

155.2 Attributes Body

```
{  
  "serverFirewallType": "{type}"  
}
```

155.2.1 Parameters

serverFirewallType
Actual type of the dedicated server firewall. Possible values - **shared** or **dedicated**.

155.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

156 ServiceAttributes-dnr.feature.domain guard

157 DomainGuard Service Attributes

Attributes used when configuring the DomainGuard service.

157.1 Contents

- 1 DomainGuard Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

157.2 Attributes Body

```
{  
  "domain": "mydomain.com",  
  "email": "test@email.com",  
  "password": "test1234"  
}
```

157.2.1 Parameters

domain - *String*
The domain name to be protected.

email - *String*
Email address of the DomainGuardian, who is the only person that will have access to modify the guarded domain

password - *String*
The password used for a DomainGuard authorization.

157.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

158 ServiceAttributes-dnr.feature.private registration

159 WhoisProtector(Private Registration) Service Attributes

Attributes used when configuring the WhoisProtector service.

159.1 Contents

- 1 WhoisProtector(Private Registration) Service Attributes
 - ◆ 1.1 Attributes Body
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 See also

159.2 Attributes Body

```
{  
  "domain": "mydomain.com",  
  "forwardEmail": "test@email.com"  
}
```

159.2.1 Parameters

domain - *String*
The domain name to be protected.

forwardEmail - *String (OPTIONAL)*
Forward email address. When messages are received at a private WhoisProtector email address, after SPAM filtering, they will be forwarded to this privacy email address.

159.3 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

160 ServiceAttributes-mcafee

161 Excel Micro Service Attributes

No service attributes are available for the Excel micro service-specifications. Attempt to setting or getting attributes for a Excel Micro will result in a **404** error.

161.1 See also

- [Service Attributes Structure](#)
- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

162 ServiceAttributesStructure

Services supported by Hostway Public Order API:

- [Web Hosting](#)
- [Domain Name Registration](#)
- [Sub-Domain Registration](#)
- [Dedicated Server](#)
- [Dedicated Server Firewall](#)
- [Exchange2010](#)
- [WhoisProtector \(Private Registration\)](#)
- [DomainGuard](#)
- [VPS Virtuozzo Server](#)
- [WebsPlanet](#)
- [Excel Micro](#)
- [Cloud Compute](#)
- [Sharepoint](#)
- [Sharepoint 2007](#)
- [Fax Over Email - User](#)
- [Fax Over Email - Number](#)
- [Online Backup - Account](#)
- [AVAS](#)

162.1 See also

- [Orders API](#)
- [Account Management API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

163 ZenDesk-SSO-Link-Per-Contact-Type:GET

164 GET /accounts/{accountId}/contacts/{contactType}/supportSSOLink

Get contact details for an account.

164.1 Contents

- 1 GET /accounts/{accountId}/contacts/{contactType}/supportSSOLink
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Successful Request
 - ◇ 1.3.2 Request for SSO link of non-verified email
 - ◇ 1.3.3 Request for non-existent account Id or contact Type
 - ◆ 1.4 See also

164.2 Request

GET /accounts/{accountId}/contacts/{contactType}/supportSSOLink

164.2.1 Request Parameters

accountId - *string*
The unique account Id.

contactType - *string*
The contact type that needs support SSO link.
Note, there are 4 available types of contact information and you can request SSO link for one of them per request: "regular", "billing", "administrator" and "technical".

164.2.2 URI Parameters

None.

164.2.3 Request Headers

Authorization - *HTTP Authorization header* [1]
The *Authentication* credentials of the client application.

164.3 Response

164.3.1 Status Code

- 200 OK
Success
- 401 Unauthorized
The request was sent without an *Authorization* header, or the user represented by the value of this header was not properly authenticated.
- 403 Forbidden
The email address of the user authenticated for this request is not verified.
- 404 Not Found
The specified account Id or contact type for this account Id does not exist or is purged.

164.3.2 Response Body

```
{
  "ssoLink" : "{ssoLink}",
  "links" : [ {
    "href" : "https://api.hostway.com/accounts/{accountId}/contacts/{contactType}/supportSSOLink/",
    "rel" : "self"
  } ]
}
```

164.3.2.1 Parameters

ssoLink - *string*
Contains a link for SSO login to ZenDesk platform.

links - *array*
Hypermedia for this resource.

164.4 Examples

164.4.1 Successful Request

Request

```
GET /accounts/{accountId}/contacts/{contactType}/supportSSOLink
Authorization: Basic {Base64-encoded username & password string}
```

Response

200 OK

```
{
  "ssoLink" : "https://hostway.zendesk.com/access/jwt?jwt={authString}",
  "links" : [ {
    "href" : "https://api.hostway.com/accounts/{accountId}/contacts/{contactType}/supportSSOLink/",
    "rel" : "self"
  } ]
}
```

164.4.2 Request for SSO link of non-verified email

Request

```
GET /accounts/{AccountId}/contacts/{ContactType}/supportSSOLink
Authorization: Basic {Base64-encoded username & password string}
```

Response

403 Forbidden

```
{
  "forbidden" : {
    "guid" : "{identifier}",
    "message" : "Email address is not verified",
    "code" : 403,
    "details" : ""
  }
}
```

164.4.3 Request for non-existent account Id or contact Type

Request

```
GET /accounts/{fakeAccountId}/contacts/{fakeContactType}/supportSSOLink
Authorization: Basic {Base64-encoded username & password string}
```

Response

404 Not Found

```
{
  "itemNotFound" : {
    "guid" : "{identifier}",
    "message" : "Resource not found",
    "code" : 404,
    "details" : ""
  }
}
```

164.5 See also

- [Account Management API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)