



## **API Documentation**

**Version 3.17**

## Table of Contents

<b>Change History</b>	<b>6</b>
<b>Contact Information</b>	<b>9</b>
<b>API Connection Details</b>	<b>9</b>
<b>API Error Handling</b>	<b>9</b>
<b>Account Balance Query</b>	<b>10</b>
<b>Domain Names &amp; DNS</b>	<b>11</b>
Domain Information Query	12
Bulk Domain Information Query	14
Password Update	17
Availability	18
List Domains	19
Bulk Availability	24
Domain Pricing	25
Register (.au specific) - Deprecated	26
Register (.uk specific) - Deprecated	28
Register (.us specific) - Deprecated	29
Register (all other extensions)	30
Is Transferrable	32
Transfer	33
Transfer (.uk specific) - Deprecated	35
Transfer Release (.uk specific)	36
Resend Transfer Approval Email	37
Resubmit Failed Transfer - Deprecated	38
Cancel Inbound Transfer	39
Approve Outbound Transfer (ack)	40
Reject Outbound Transfer	41
Renew	42
RGP Restore Domain	43
Update Name Servers	44
Updating Contacts	45
Resend Registrant Update Email	46
Cancel Pending Registrant Update	47
List Contacts	48
List ID Protected Contacts	49
Get Raw Contacts	50
Add Registry Host	51
Delete Registry Host	52
Add Registry Host IP Address	53
Delete Registry Host IP Address	54
Registry Host Information	55
List All Registry Hosts	56
Transfer Lock	57
Transfer Unlock	58
Domain Renew Required Check	59
Max Years Domain Can Be Renewed For	60
List Available Domain Extensions	61
.XXX Membership Details	62
Retrieve .US Nexus Data	63

.AU Change of Registrant Request	64
Cancel Change Of Registrant Request	65
Add DNSSEC Record	66
Remove DNSSEC Record	67
List DNSSEC Entries	68
DNSSEC Information	69
Extension Options	70
Determine Domain Is Renewable	72
Transferred Away	73
Resend ICANN Registrant Verification Email	74
Add DNS Zone	75
Delete DNS Zone	76
Add DNS Record	77
Update DNS Record	78
Delete DNS Record	79
Get DNS Record	80
List DNS Zone	81
Add Email Forward	82
Delete Email Forward	83
List Email Forwards	84
Add Simple URL Forward	85
Delete Simple URL Forward	86
List Simple URL Forwards	87
Enable ID Privacy Protection	88
Disable ID Privacy Protection	89
Enable Automatic Renewal	90
Disable Automatic Renewal	91
Lookup ABN/ACN/RBN Information	92
Generate .au Eligibility from Business Number	93
Get Eligibility Fields	95
List Domain Categories	98
Create Domain Category	99
Update Domain Category	100
Remove Domain Category	101
Assign Domain Category	102
Unassign Domain Category	103
Initiate Domain Registrant Correction	104
Check Domain Registrant Correction	105
Resend Domain Registrant Correction Email	106
<b>Hosting Services</b>	<b>107</b>
List Hosting Services	108
Hosting Service Query	110
Bulk Hosting Service Query	112
Purchase Hosting Service	114
Suspend Hosting Service	115
Unsuspend Hosting Service	116
List Hosting Packages	117
Change Hosting Password	118
Change Hosting Package	119

Enable Temporary URL	120
Disable Temporary URL	121
Check IP Address for Firewall Block	122
Unblock IP Address from Firewall Block	123
Recreate Hosting Service	124
Terminate Hosting Service	125
Generate Hosting Login Link	126
<b>SSL Certificates</b>	<b>127</b>
Retrieve SSL Product Information & Pricing	128
Query Certificate Status	129
Query Certificate Status (simple)	130
Generate Certificate Signing Request (CSR)	131
Decode Certificate Signing Request (CSR)	132
Purchase New SSL Certificate	133
Reissue SSL Certificate	134
Cancel SSL Certificate	135
Renew SSL Certificate	136
Resend Approval Email	137
Resend Completion Certificate Email	138
Force Check TXT Record for DNS Domain Validation	139
List All Certificates	140
Get Domain Beacon	141
Check Domain Beacon	142
<b>Services &amp; Product Information</b>	<b>143</b>
SSL Certificate Product IDs	144
Hosting Plans Information	145
Hosting Locations Information	146
DNS Configuration Information	147
Country List	148
State List	152
Domain Registration Information (All Extensions)	153
.AU Domain Registration Information	154
.US Domain Registration Information	155
ICANN Whois Data Verification Statuses	156
<b>API Error Messages</b>	<b>157</b>
API Error Messages - Domain Specific	158
API Error Messages - DNS Specific	159
<b>API Usage Examples</b>	<b>160</b>
Example - .AU Domain Registrations	161
Example - Name Server Update	162
Example - SSL Certificate Purchase	163
Example - DNS Record ID Calculation	164
<b>Microsoft 365</b>	<b>165</b>
Create a Client	167
Updating Client Details	168
Getting Client Details	169
Listing all Clients	171
Listing Subscriptions Available for Purchase	172
Purchase a Subscription/Addon	173

Get Subscription Details	174
Update Subscription Quantity	175
Suspend Subscription	176
Unsuspend Subscription	177
List Client Subscriptions	178
Terminate Subscription	180

# Change History

Date	Description	Version
23-04-2026	<ul style="list-style-type: none"> <li>Removed Legacy SMS Commands</li> </ul>	3.17
04-03-2026	<ul style="list-style-type: none"> <li>Add new endpoint for cancelling a change of registrant (COR) request</li> <li>Add new endpoint to check pending domain registrant corrections</li> </ul>	3.16
10-02-2026	<ul style="list-style-type: none"> <li>Add new endpoint for initiate domain registrant correction</li> <li>Add new endpoint for resending domain registrant correction email</li> </ul>	3.15
01-01-2026	<ul style="list-style-type: none"> <li>Correction of response for listContacts from [company] to [organisation]</li> <li>Correction of response for listMailForwards from [records] to [forwards]</li> </ul>	3.14
27-10-2025	<ul style="list-style-type: none"> <li>Add Missing Command for getDNSRecord</li> </ul>	3.13
03-10-2025	<ul style="list-style-type: none"> <li>Add Missing Command for updateDNSRecord</li> </ul>	3.12
14-06-2024	<ul style="list-style-type: none"> <li>Add new endpoints for domain categories               <ul style="list-style-type: none"> <li>listDomainCategories</li> <li>createDomainCategory</li> <li>updateDomainCategory</li> <li>removeDomaincategory</li> <li>assignDomainCategory</li> <li>unassignDomainCategory</li> </ul> </li> <li>Add extra 'categories' field to related result for the following endpoints               <ul style="list-style-type: none"> <li>listDomains</li> <li>bulkDomainInfo</li> <li>domainInfo</li> </ul> </li> </ul>	3.11
17-05-2024	<ul style="list-style-type: none"> <li>Update 'domainInfo' and 'bulkDomainInfo' to now return additional information for AU domains regarding the status of their eligibility information.</li> </ul>	3.10
07-09-2023	<ul style="list-style-type: none"> <li>Addition of new Microsoft 365 commands               <ul style="list-style-type: none"> <li>subscriptionCreateClient</li> <li>subscriptionUpdateClient</li> <li>subscriptionGetClient</li> <li>subscriptionListClients</li> <li>subscriptionListPurchasable</li> <li>subscriptionPurchase</li> <li>subscriptionGetDetails</li> <li>subscriptionUpdateQuantity</li> <li>subscriptionSuspend</li> <li>subscriptionUnsuspend</li> <li>subscriptionListClientSubscriptions</li> <li>subscriptionTerminate</li> </ul> </li> <li>Update 'updateContacts' contact set requirements to be clearer</li> </ul>	3.9
14-11-2022	<ul style="list-style-type: none"> <li>Updated 'listDNSZone' to now return return the correct ID               <ul style="list-style-type: none"> <li>Included an Example of how this ID is calculated</li> </ul> </li> <li>Update 'getDomainPricing' to include sale pricing</li> <li>Addition of entityTypeCode property on 'businessCheckRegistration'</li> <li>Addition of cannotRestoreAfter property on 'getDomainExtensionOptions'</li> <li>Addition of deletesAfter property on 'getDomainExtensionOptions'</li> <li>Addition of canRestore property on 'getDomainExtensionOptions'</li> <li>Addition of subjectAltNames property on 'SSL_decodeCSR'</li> <li>New commands               <ul style="list-style-type: none"> <li>rawDomainContacts</li> <li>hostingGetLogin</li> <li>isDomainTransferrable</li> </ul> </li> <li>Removed the 'massChangeTTL' command</li> </ul>	3.8
10-06-2020	<ul style="list-style-type: none"> <li>Added support for Email Hosting               <ul style="list-style-type: none"> <li>Addition of the 'identifier' parameter to deprecate the 'hoid' field. This field accepts either the 'hoid' or the email of the email hosting account. This is only found on the following API mentioned below..</li> <li>Updated Purchase Hosting Service to support Email Hosting</li> <li>Updated Suspend Hosting Service to support Email Hosting</li> <li>Updated Unsuspend Hosting Service to support Email Hosting</li> <li>Updated Change Hosting Package to support Email Hosting</li> <li>Updated Change Hosting Password to support Email Hosting</li> <li>Updated Terminate Hosting Service to support Email Hosting</li> </ul> </li> <li>Deprecated 'hoid' field</li> <li>Added 'hostingListPackages' command to retrieve a list of configured hosting plans</li> </ul>	3.7
09-04-2020	<ul style="list-style-type: none"> <li>Formatting changes to the table of contents to correct headings to make navigation easier</li> </ul>	3.6

03-03-2020	<ul style="list-style-type: none"> <li>Updated 'getTransferredAwayDomains' to now return 'domain_id'</li> <li>Updates to support contact/transfer eligibility information, includes; <ul style="list-style-type: none"> <li>Addition of 'type' parameter to 'getDomainEligibilityDetails' request</li> <li>Addition of 'eligibility' parameter to 'transferDomain' request</li> </ul> </li> <li>Updates to request to support post-registration/post-transfer actions; <ul style="list-style-type: none"> <li>Addition of 'dnsConfig' and 'nameServers' parameters to 'transferDomain' requests</li> <li>Addition of 'categories' and 'transferLock' parameters to 'domainRegister' and 'transferDomain' requests</li> </ul> </li> </ul>	3.5
10-02-2020	<ul style="list-style-type: none"> <li>Deprecated domainRegisterAU</li> <li>Deprecated domainRegisterUK</li> <li>Deprecated domainRegisterUS</li> <li>Deprecated transferDomainUK</li> <li>Deprecated resubmitFailedTransfer</li> <li>Updates to support premium domain names, includes; <ul style="list-style-type: none"> <li>Addition of 'command' and 'years' parameters to 'checkDomain' and 'bulkCheckDomain' requests</li> <li>Addition of 'premium' property to 'checkDomain' and 'bulkCheckDomain' responses</li> <li>Addition of 'premium' and 'costPrice' parameters to 'domainRegister' request</li> <li>Addition of 'premium' and 'costPrice' parameters to 'renewDomain' request</li> <li>Addition of 'premium' and 'costPrice' parameters to 'transferDomain' request</li> <li>Addition of 'premium' parameter to the 'restoreDomain' request</li> </ul> </li> <li>New commands <ul style="list-style-type: none"> <li>generateAuEligibility</li> <li>getDomainEligibilityFields</li> <li>transferCancel</li> </ul> </li> <li>Updated 'hostingPurchaseService' parameters to reflect the current state of the command</li> </ul>	3.4
19-07-2018	<ul style="list-style-type: none"> <li>Addition of createdDate property on 'domainInfo' and 'bulkDomainInfo' commands</li> <li>Addition of years property that can be supplied when using 'initiateAUCOR' command</li> </ul>	3.3
02-07-2018	<ul style="list-style-type: none"> <li>Inclusion of 'listProtectedContacts' command</li> <li>Inclusion of 'hostingEnableTempUri' command</li> <li>Inclusion of 'hostingDisableTempUri' command</li> <li>Inclusion of 'hostingCheckFirewall' command</li> <li>Inclusion of 'hostingUnblockFirewall' command</li> <li>Update of Synergy Wholesale branding</li> </ul>	3.2
05-04-2018	<ul style="list-style-type: none"> <li>'domainTransferUK' command can now accept authInfo variable (domain name password) for internal transfers only (between Synergy Wholesale partner accounts)</li> </ul>	3.1
15-03-2018	<ul style="list-style-type: none"> <li>Addition of 'nz_privacy' variable in 'updateContact' command</li> <li>Modification of response in 'listContacts' command to show 'nz_privacy' key value</li> </ul>	3.0
06-02-2018	<ul style="list-style-type: none"> <li>Inclusion of 'autoRenew' field on 'domainRegister' command</li> <li>Inclusion of 'autoRenew' field on 'domainRegisterUS' command</li> <li>Inclusion of 'autoRenew' field on 'domainRegisterUK' command</li> <li>Inclusion of 'autoRenew' field on 'domainRegisterAU' command</li> <li>Addition of 'getSSLPricing' command</li> </ul>	2.9
29-11-2017	<ul style="list-style-type: none"> <li>Addition of 'listDomains' command</li> <li>Addition of 'listHosting' command</li> <li>Addition of 'bulkHostingInfo' command</li> <li>Separation of API product functions and controls into clearly defined sections within documentation</li> </ul>	2.8
29-11-2017	<ul style="list-style-type: none"> <li>Inclusion of 'specialConditionsAgree' boolean field on 'domainRegister' command</li> <li>Addition of 'resendRegistrantUpdateEmails' command</li> <li>Addition of 'cancelRegistrantUpdate' command</li> <li>Addition of 'transferOutboundApprove' command</li> <li>Addition of 'transferReject' command</li> <li>Addition of 'SSL_checkTxtCodes' command</li> <li>Updated Domain Registration Information section to include further information about 'speicalConditionsAgree' optional boolean when using 'domainRegister' command</li> </ul>	2.7
17-07-2017	<ul style="list-style-type: none"> <li>Inclusion of additional request variables in getTransferredAwayDomains command</li> <li>Addition of reseller response in getTransferredAwayDomains command</li> </ul>	2.6
07-07-2017	<ul style="list-style-type: none"> <li>Addition of passwordRequirements response for 'getDomainExtensionOptions' command</li> <li>Addition of 'bulkDomainInfo' command</li> </ul>	2.5
02-08-2016	<ul style="list-style-type: none"> <li>Addition of 'domainRenewRequired' command</li> <li>Addition of 'maxYearsCanRenewFor' command</li> <li>Addition of 'listAvailableDomainExtensions' command</li> </ul>	2.4
20-06-2016	<ul style="list-style-type: none"> <li>Addition of new results returned as part of a 'domainInfo' function specific for .AU domain names.</li> <li>Correction of incorrectly labelled variable for setting DNS configuration (incorrectly labelled as <i>dnsConfigType</i> when correct variable is <i>dnsConfig</i>).</li> <li>Removal of <i>DNSSECSigned</i> variable.</li> <li>Introduction of <i>DSDData</i> variable which replaces the unused <i>DNSSECSigned</i> variable.</li> </ul>	2.3
01-05-2016	<ul style="list-style-type: none"> <li>Clarification on requirements for nexusCategory and appPurpose for .US domain names on updateContact command</li> </ul>	2.2

	<ul style="list-style-type: none"> <li>• Clarification on required country information for contact creation or update, where a nexusCategory of C31 or C32 is provided</li> </ul>	
22-02-2016	<ul style="list-style-type: none"> <li>• Added 'getDomainPricing' function</li> <li>• Added 'restoreDomain' function</li> <li>• Updated 'updateNameServers' function to include the new <i>skipDefaultARecords</i> parameter</li> <li>• Updated 'getDomainExtensionOptions' function to include new returned <i>availableContacts</i> and <i>whoisVerification</i> parameters</li> <li>• Updated 'checkDomain' function to include new returned <i>costPrice</i> parameter</li> <li>• Updated 'bulkCheckDomain' function to include new returned <i>costPrice</i> parameter</li> <li>• Removal of 'root@domainname.com' email address on the SSL certificate documentation</li> <li>• Formatting changes to Table of Contents</li> </ul>	2.1
12-01-2016	<ul style="list-style-type: none"> <li>• Correction of 'bulkcheckDomain' to the correct case-sensitive command 'bulkCheckDomain'</li> </ul>	2.0
15-12-2015	<ul style="list-style-type: none"> <li>• Addition of status field, timestamp and registrar that is returned as part of the updated response to getTransferredAwayDomains</li> </ul>	1.9
05-02-2015	<ul style="list-style-type: none"> <li>• New functions added for DNSSEC for .au domain names <ul style="list-style-type: none"> <li>• DNSSECAddDS</li> <li>• DNSSECRemoveDS</li> <li>• DNSSECListDS</li> </ul> </li> <li>• Update to getDomainExtensionOptions function to include DNSSECAvailable</li> <li>• Update to domainInfo function to include DNSSECSigned</li> </ul>	1.8
22-12-2014	<ul style="list-style-type: none"> <li>• Removal of the 'hosts' variable to be sent with listAllHosts function is not required</li> </ul>	1.7
18-11-2014	<ul style="list-style-type: none"> <li>• Additional examples provided for .au domain name registration for "Example - .AU Domain Registrations"</li> <li>• Correction to column titles and associated field names added for ".AU Domain Registration Information"</li> </ul>	1.6
14-07-2014	<ul style="list-style-type: none"> <li>• Update to hostingGetService function to provide the service password and if a dedicated IP address is signed</li> <li>• New function for hosting service termination</li> <li>• New function to recreate a hosting service without canceling the billing</li> </ul>	1.5
08-07-2014	<ul style="list-style-type: none"> <li>• SMS functions have now been included into the merged API documentation</li> </ul>	1.4
07-07-2014	<ul style="list-style-type: none"> <li>• Merged API documentation for domains, SSL, hosting and SMS</li> <li>• API URL updated for new location under Synergy Wholesale</li> </ul>	1.3
28-05-2014	<ul style="list-style-type: none"> <li>• Additional statuses provided for the ICANN Whois Verification Process</li> </ul>	1.2
27-05-2014	<ul style="list-style-type: none"> <li>• Correction to ICANN Whois Verification Email Resend value of 'domain' to 'domainName'</li> </ul>	1.1
16-05-2014	<ul style="list-style-type: none"> <li>• Synergy Wholesale Domain API Document Creation</li> </ul>	1.0

## Contact Information

Questions or problems relating to our API should be directed to us using the below contact information.

<b>Email</b>	Submit a [New Request] via the Synergy Wholesale system [Support Centre]
<b>Phone</b>	+61 3 8399 9483

## API Connection Details

Our API uses the SOAP/WSDL connection method for customers to interact with our wholesale system. For each command passed to our API you must issue the connection using the below details.

Our API is case sensitive. All fields of "command" and "variable" must be passed to the API exactly as shown in this document. Eg. "apiKey" - you must have a capital "K" for the key to be recognised by our API.

**Note:** All customers must provide us with their IP address(es) of the server which will connect to the API.

<b>API Location</b>	<a href="https://api.synergywholesale.com">https://api.synergywholesale.com</a>
---------------------	---

## API Error Handling

Our API will also return a 'status' for every command as well as an 'errorMessage' for any commands that failed to execute. API error messages are included with this document to assist with reasons for failed requests.

To assist with debugging errors we also provide the server side API logging via the wholesale system interface.

Error Response		
Variable	Data Type	Description
status	String	Status is returned to advise if the function completed successfully or failed to execute
errorMessage	String	Contains a brief description of why the command failed to execute

Example of failed response
[status] => ERR_LOGIN_FAILED [errorMessage] => Unable to login to wholesale system

# Account Balance Query

The following command allows you to obtain the account balance.

**Command:** balanceQuery

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
balance	String	"balance" will contain your account balance (eg. 100.00)

Example of successful account balance
[status] => OK [balance] => 802.25

Example of failed account balance
[status] => ERR_LOGIN_FAILED [errorMessage] => Unable to login to wholesale system

# **Domain Names & DNS**

The following commands allow for acquisition and management of domain names, their assigned contacts and other DNS related functionality

# Domain Information Query

The following command allows you to obtain information for a domain name.

**Command:** domainInfo

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
domainName	String	Domain name of the query you perform
domain_status	String	Status of domain name
domain_expiry	String	Domain name expiry date
createdDate	String	The date of creation for the domain name
nameServers	Array	Name servers assigned to the domain name
dnsConfig	String	Refer to "DNS Configuration Information" section in this document
dnsConfigName	String	DNS configuration full description
domainPassword	String	Password for the domain name (AuthInfo/EPP)
bulkInProgress	String	Determines if there is a bulk update in progress
idProtect	String	Determines if ID Protection is enabled/disabled
autoRenew	String	Determines if Auto Renewal is enabled/disabled
auRegistrantIDType	String	Provides the registrant type of ABN or ACN (.AU domains only)
auRegistrantID	String	Provides the registrant ABN or ACN number (.AU domains only)
auRegistrantName	String	Provides the registrant name (.AU domains only)
auEligibilityName	String	Provides the registrant eligibility name (.AU domains only)
auEligibilityID	String	Provides the registrant eligibility ID (.AU domains only)
auEligibilityType	String	Provides the registrant eligibility type (.AU domains only)
auEligibilityIDType	String	Provides the registrant eligibility type ID (.AU domains only)
auPolicyID	String	Provides the registration policy ID (.AU domains only)
auPolicyIDDesc	String	Provides a description on what the policy ID that is set means (.AU domains only)
icannStatus	String	Status of Registrant Verification, refer to the section in this document of "ICANN Whois Data Verification Statuses"
icannVerificationDateEnd	String	If the Registrant contact data isn't verified by this timestamp the domain will be placed in a ClientHold status which will stop DNS resolution (websites and/or email services will cease).
DSDData	Array	Note: DSDData is only returned if there is DNSSEC Data associated with the domain name. It will return any DNSSEC Data assigned in an array format.

au_valid_eligibility	Boolean	Returns if the currently set eligibility information is found to be valid or not. (.AU domains only)
au_eligibility_last_check	String	The date in ISO8601 short format (YYYY-MM-DD HH:ii:ss) of the last time the validity of the eligibility data was checked. (.AU domains only)
category	Array	Array containing details of the categories that this domain has been assigned to
id	String	The ID of the domain category
name	String	The name of the domain category
note	String	The extra note of the domain category
createdDate	dateTime	The date and time that the domain category was created

#### Example of successful domain information query

```
[domainName] => synergywholesale.com
[domain_status] => clientTransferProhibited
[domain_expiry] => 2022-01-17 16:31:47
[domain_create] => 2000-06-10 04:58:24
[nameServers] => Array
(
  [0] => NS1.HOST-SERVICES.US
  [1] => NS2.HOST-SERVICES.US
)

[dnsConfig] => 1
[dnsConfigName] => Custom
[bulkInProgress] => 0
[domainPassword] => *****
[status] => OK
[idProtect] => Disabled
[autoRenew] => off
[icannVerificationDateEnd] => 2014-05-31 23:59:59
[icannStatus] => Pending Verification
[DSDData] => Array
(
  [keyTag] => 9885
  [Algoirthm] => 5
  [Digest] => 476XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
  [DigestType] => 1
  [UUID] => 87xxx5xxx4
)
[categories] => Array
(
  [0] => Array
  (
    [id] =>
    [name] => Category Test One
    [note] => Testing Note One
    [createdDate] => 2024-01-01 00:00:00
  )

  [1] => Array
  (
    [id] =>
    [name] => Category Test Two
    [note] => Testing Note Two
    [createdDate] => 2024-03-01 00:00:00
  )
)
)
```

#### Example of non-existent domain information query

```
[status] => ERR_DOMAININFO_FAILED
[errorMessage] => Domain Info Failed - Domain Does Not Exist
[domainName] => nonexistentdomain.com.au
[domain_status] => Domain does not exist
```

# Bulk Domain Information Query

The following command allows you to obtain information from multiple domain names through a single bulk action.

**Command:** bulkDomainInfo

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainList	Mandatory	An array containing each domain name you would like information returned on

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
domainList	Array	Contains the nested domain information for each domain. Please see 'domainInfo' command for information returned for each domain name in the array

## Example of successful bulk domain information query

```
[domainList] => Array
(
  [0] => stdClass Object
  (
    [status] => OK
    [domainName] => hellosynergywholesale.com.au
    [domainPassword] => v[0d^gnDs3]5
    [domainStatus] => ok
    [domain_expiry] => 2017-06-14 11:09:13
    [domain_create] => 2000-06-10 04:58:24
    [dnsConfig] => 0
    [nameServers] => Array
    (
      [0] => ns1.nameserver.net.au
      [1] => ns2.nameserver.net.au
      [2] => ns3.nameserver.net.au
    )
    [categories] => Array
    (
      [0] => Array
      (
        [id] =>
        [name] => Category Test One
        [note] => Testing Note One
        [createdDate] => 2024-01-01 00:00:00
      )
      [1] => Array
      (
        [id] =>
        [name] => Category Test Two
        [note] => Testing Note Two
        [createdDate] => 2024-03-01 00:00:00
      )
    )
    [contacts] => stdClass Object
    (
      [registrant] => stdClass Object
      (
        [company] => Synergy Wholesale
        [firstname] => Johnny
        [lastname] => Appleseed
        [address1] => 1 Infinite Loop
        [address2] =>
        [address3] =>
        [suburb] => Cupertino
        [state] => CA
        [country] => USA
        [postcode] => 95014
        [phone] => +1.1112346643
      )
    )
  )
)
```

```
        [fax] =>
        [email] => email@address.com
    )
    [tech] => stdClass Object
    (
        [company] =>
        [firstname] => Johnny
        [lastname] => Appleseed
        [address1] => 1 Infinite Loop
        [address2] =>
        [address3] =>
        [suburb] => Cupertino
        [state] => CA
        [country] => USA
        [postcode] => 95014
        [phone] => +1.1112346643
        [fax] =>
        [email] => email@address.com
    )
)
[1] => stdClass Object
(
    [status] => OK
    [domainName] => helloworld.com
    [domainPassword] => Uj{pxITdT_}o6U3
    [domainStatus] => ok
    [domain_expiry] => 2017-06-14 11:10:06
    [domain_create] => 2000-06-10 04:58:24
    [dnsConfig] => 0
    [nameServers] => Array
    (
        [0] => ns1.nameserver.net.au
        [1] => ns2.nameserver.net.au
        [2] => ns3.nameserver.net.au
    )
    [categories] => Array
    (
        [0] => Array
        (
            [id] =>
            [name] => Category Test One
            [note] => Testing Note One
            [createdDate] => 2024-01-01 00:00:00
        )
        [1] => Array
        (
            [id] =>
            [name] => Category Test Two
            [note] => Testing Note Two
            [createdDate] => 2024-03-01 00:00:00
        )
    )
    [contacts] => stdClass Object
    (
        [admin] => stdClass Object
        (
            [company] =>
            [firstname] => Johnny
            [lastname] => Appleseed
            [address1] => 1 Infinite Loop
            [address2] =>
            [address3] =>
            [suburb] => Cupertino
            [state] => CA
            [country] => USA
            [postcode] => 95014
            [phone] => +1.1112346643
            [fax] =>
            [email] => email@address.com
        )
        [tech] => stdClass Object
        (
            [company] =>
            [firstname] => Johnny
            [lastname] => Appleseed
            [address1] => 1 Infinite Loop
            [address2] =>
            [address3] =>
            [suburb] => Cupertino
            [state] => CA
        )
    )
)
```

```
        [country] => USA
        [postcode] => 95014
        [phone] => +1.1112346643
        [fax] =>
        [email] => email@address.com
    )
    [billing] => stdClass Object
    (
        [company] =>
        [firstname] => Johnny
        [lastname] => Appleseed
        [address1] => 1 Infinite Loop
        [address2] =>
        [address3] =>
        [suburb] => Cupertino
        [state] => CA
        [country] => USA
        [postcode] => 95014
        [phone] => +1.1112346643
        [fax] =>
        [email] => email@address.com
    )
    [registrant] => stdClass Object
    (
        [company] =>
        [firstname] => Johnny
        [lastname] => Appleseed
        [address1] => 1 Infinite Loop
        [address2] =>
        [address3] =>
        [suburb] => Cupertino
        [state] => CA
        [country] => USA
        [postcode] => 95014
        [phone] => +1.1112346643
        [fax] =>
        [email] => email@address.com
    )
)
)
```

#### Example of failed bulk domain information query

#### Example of total bulk domain information lookup failure

```
[status] => ERR_OCCURRED
[errorMessage] => Failed to perform bulk lookup. Please try again later.
```

#### Example of a single domain name not existing in a bulk domain information lookup

```
[status] => OK
[domainList] => Array
(
    [0] => stdClass Object
    (
        [status] => ERR_DOMAIN_NOT_FOUND
        [errorMessage] => Domain name does not exist.
    )
)
```

# Password Update

The following command allows you to update the password of a domain name.

**Command:** updateDomainPassword

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name you wish to update the password
newPassword	Mandatory	New password you wish to update

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful domain password update
[status] => OK [errorMessage] => Domain password has been updated

Example of failed domain password update
[statusCode] => ERR_UPDATEDDOMAINPASSWORD_FAILED [reason] => Unable to update the domain password

# Availability

The following command allows you to check the availability of a domain name. All domain extensions provided are available to be checked using this function.

**Command:** checkDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
command	Optional	Used to retrieve the pricing for the specified command. Provide one of the following values; "create", "transfer", "renew", "restore". Defaults to "create" when not variable is not provided.
years	Optional	Used to retrieve the pricing for a specified number of years. Defaults to "1" when the variable is not provided.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the registration on the absolute minimum period
premium	Boolean	Boolean value to indicate if the domain name is premium

## Example of successful availability check

```
[status] => AVAILABLE  
[costPrice] => 19.95  
[premium] => false
```

## Example of unavailable domain check

```
[status] => UNAVAILABLE
```

## List Domains

The following command returns a paginated domain information for your Synergy Wholesale partner account. **Note:** There is a maximum of 500 domains returned per page.

**Command:** listDomains

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
page	Optional	An integer specifying the page you would like to return
limit	Optional	An integer specifying the result limit (500 are returned if no value supplied)
status	Optional	Allows you to filter domain names by status ('ok', 'clienhold', 'dropped', 'transferredaway', 'deleted', 'inactive', 'clientTransferProhibited', 'cilentUpdatedProhibited', 'pendingDelete', 'policyDelete', 'redemption', etc.) <b>Note:</b> Only one value can be supplied at a time.

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the failed command
domainList	array	Contains the nested domain info for each domain
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the failed command
domainName	string	Domain name of the query you perform
domain_status	string	Status of domain name
domain_expiry	string	Domain name expiry date
nameServers	array	Name servers assigned to the domain name
dnsConfig	string	Refer to "DNS Configuration Information" section in this document
dnsConfigName	string	DNS configuration full description
domainPassword	string	Password for the domain name (AuthInfo/EPP)
bulkInProgress	string	Determines if there is a bulk update in progress
idProtect	string	Determines if ID Protection is enabled/disabled
autoRenew	string	Determines if Auto Renewal is enabled/disabled
auRegistrantIDType	string	Provides the registrant type of ABN or ACN (.AU domains only)
auRegistrantID	string	Provides the registrant ABN or ACN number (.AU domains only)
auRegistrantName	string	Provides the registrant name (.AU domains only)
auEligibilityName	string	Provides the registrant eligibility name (.AU domains only)
auEligibilityID	string	Provides the registrant eligibility ID (.AU domains only)
auEligibilityType	string	Provides the registrant eligibility type (.AU domains only)
auEligibilityIDType	string	Provides the registrant eligibility type ID (.AU domains only)
auPolicyID	string	Provides the registration policy ID (.AU domains only)

auPolicyIDDesc	string	Provides a description on what the policy ID that is set means (.AU domains only)
icannStatus	string	Status of Registrant Verification, refer to the section in this document of "ICANN Whois Data Verification Statuses"
icannVerificationDateEnd	string	If the Registrant contact data isn't verified by this timestamp the domain will be placed in a ClientHold status which will stop DNS resolution (websites and/or email services will cease).
DSDData	array	Note: DSDData is only returned if there is DNSSEC Data associated with the domain name. It will return any DNSSEC Data assigned in an array
category	Array	Array containing details of the categories that this domain has been assigned to
id	String	The ID of the domain category
name	String	The name of the domain category
note	String	The extra note of the domain category
createdDate	dateTime	The date and time that the domain category was created

#### Example of successful list domains command

```
[status] => OK
[limit] => 5
[page] => 1
[domainList] => Array
(
  [0] => stdClass Object
  (
    [status] => OK
    [domainName] => thesuperpotatoclub.net.au
    [domainPassword] => v[0d^gnDs3]5}
    [domainStatus] => ok
    [domain_expiry] => 2017-06-14 11:09:13
    [dnsConfigName] => Custom
    [bulkInProgress] => 0
    [icannVerificationDateEnd] => 2014-05-31 23:59:59
    [icannStatus] => Pending Verification
    [idProtect] => Disabled
    [autoRenew] => off
    [nameServers] => Array
    (
      [0] => ns1.nameserver.net.au
      [1] => ns2.nameserver.net.au
      [2] => ns3.nameserver.net.au
    )
    [categories] => Array
    (
      [0] => Array
      (
        [id] =>
        [name] => Category Test One
        [note] => Testing Note One
        [createdDate] => 2024-01-01 00:00:00
      )
      [1] => Array
      (
        [id] =>
        [name] => Category Test Two
        [note] => Testing Note Two
        [createdDate] => 2024-03-01 00:00:00
      )
    )
    [contacts] => stdClass Object
    (
      [registrant] => stdClass Object
      (
        [company] =>
        [firstname] => Johnny
        [lastname] => Appleseed
        [address1] => 1 Infinite Loop
        [address2] =>
        [address3] =>
        [suburb] => Cupertino
        [state] => CA
        [country] => USA
        [postcode] => 95014
        [phone] => +1.1112346643
      )
    )
  )
)
```

```

    [fax] =>
    [email] => email@address.com
  )

  [tech] => stdClass Object
  (
    [company] =>
    [firstname] => Johnny
    [lastname] => Appleseed
    [address1] => 1 Infinite Loop
    [address2] =>
    [address3] =>
    [suburb] => Cupertino
    [state] => CA
    [country] => USA
    [postcode] => 95014
    [phone] => +1.1112346643
    [fax] =>
    [email] => email@address.com
  )
)

[DSDData] => Array
(
  [keyTag] => 9885
  [Algorithm] => 5
  [Digest] => 476XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
  [DigestType] => 1
  [UUID] => 87xxx5xxx4
)
[auRegistrantName] => SYNERGY WHOLESALE PTY LTD
[auRegistrantID] => 17 165 245 901
[auRegistrantIDType] => ABN
[auEligibilityName] =>
[auEligibilityID] =>
[auEligibilityType] => Company
[auEligibilityIDType] =>
[auPolicyID] => 1
[auPolicyIDDesc] => Domain name is an exact match, abbreviation or acronym of the registrant's name or trade mark
)

[1] => stdClass Object
(
  [status] => OK
  [domainName] => thesuperpotatoclub.com
  [domainPassword] => U}{pxlTdT_}o6U3
  [domainStatus] => ok
  [domain_expiry] => 2017-06-14 11:10:06
  [dnsConfig] => 0
  [dnsConfigName] => Custom
  [bulkInProgress] => 0
  [icannVerificationDateEnd] => 2014-05-31 23:59:59
  [icannStatus] => Pending Verification
  [idProtect] => Disabled
  [autoRenew] => off
  [nameServers] => Array
  (
    [0] => ns1.nameserver.net.au
    [1] => ns2.nameserver.net.au
    [2] => ns3.nameserver.net.au
  )
  [categories] => Array
  (
    [0] => Array
    (
      [id] =>
      [name] => Category Test One
      [note] => Testing Note One
      [createdDate] => 2024-01-01 00:00:00
    )

    [1] => Array
    (
      [id] =>
      [name] => Category Test Two
      [note] => Testing Note Two
      [createdDate] => 2024-03-01 00:00:00
    )
  )
  [contacts] => stdClass Object
  (
    [admin] => stdClass Object
    (
      [company] =>
      [firstname] => Johnny
      [lastname] => Appleseed
      [address1] => 1 Infinite Loop
      [address2] =>
    )
  )
)

```

```
[address3] =>
[suburb] => Cupertino
[state] => CA
[country] => USA
[postcode] => 95014
[phone] => +1.1112346643
[fax] =>
[email] => email@address.com
)
```

```
[tech] => stdClass Object
```

```
(
  [company] =>
  [firstname] => Johnny
  [lastname] => Appleseed
  [address1] => 1 Infinite Loop
  [address2] =>
  [address3] =>
  [suburb] => Cupertino
  [state] => CA
  [country] => USA
  [postcode] => 95014
  [phone] => +1.1112346643
  [fax] =>
  [email] => email@address.com
)
```

```
[billing] => stdClass Object
```

```
(
  [company] =>
  [firstname] => Johnny
  [lastname] => Appleseed
  [address1] => 1 Infinite Loop
  [address2] =>
  [address3] =>
  [suburb] => Cupertino
  [state] => CA
  [country] => USA
  [postcode] => 95014
  [phone] => +1.1112346643
  [fax] =>
  [email] => email@address.com
)
```

```
[registrant] => stdClass Object
```

```
(
  [company] =>
  [firstname] => Johnny
  [lastname] => Appleseed
  [address1] => 1 Infinite Loop
  [address2] =>
  [address3] =>
  [suburb] => Cupertino
  [state] => CA
  [country] => USA
  [postcode] => 95014
  [phone] => +1.1112346643
  [fax] =>
  [email] => email@address.com
)
```

```
)
```

```
)
```

```
[2] => stdClass Object
```

```
(
  [status] => OK_PENDING_MANUAL
  [domainName] => thepotatoclub.com.au
)
```

```
[3] => stdClass Object
```

```
(
  [status] => OK_TRANSFER_PENDING
  [transfer_status] => Pending Registry Completion
  [transfer_completed_by] => 2017-07-15 08:56:24
  [email] => potato@webmail.com
  [domainName] => ilovethepotatolyf.com.au
)
```

```
[4] => stdClass Object
```

```
(
  [status] => OK_PENDING_COR
  [errorMessage] => Awaiting contact information from the NEW registrant.
  [domainName] => mypotato.net.au
)
```

```
)
```

#### Example of failed list domains command where a domain does not exist

```
[status] => OK
[limit] => 5
[page] => 1
[domainList] => Array
(
  [0] => stdClass Object
    (
      [domainName] => thepotatoclub.org
      [status] => ERR_DOMAIN_NOT_FOUND
      [errorMessage] => Domain name does not exist
    )
)
```

#### Example of failed list domains command

```
[status] => ERR_OCCURED
[errorMessage] => Failed to list domains. Please try again later
```

# Bulk Availability

The following command allows you to check the availability of up to 30 domain names. All domain extensions provided are available to be checked using this function.

**Command:** bulkCheckDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainList	Mandatory	Array of domain names with extensions to check for availability
command	Optional	Used to retrieve the pricing for the specified command. Provide one of the following values; "create", "transfer", "renew", "restore". Defaults to "create" when no variable is not provided.
years	Optional	Used to retrieve the pricing for a specified number of years. Defaults to "1" when the variable is not provided.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
domainList	Array	List of domain names and the availability status. 0 - Unavailable, 1 - Available
costPrice	String	Amount charged to account for the registration on the absolute minimum period. If domain name is not available, the <i>costPrice</i> will not be returned
premium	Boolean	Boolean value to indicate if the domain name is premium, the <i>premium</i> parameter will not be returned if the domain is unavailable.

## Example of bulk availability check

```
[status] => ok
[errorMessage] => Bulk Check Completed Successfully
[domainList] => Array
(
  [0] => Array
  (
    [domain] => testdomain.com
    [available] => 0
  )
  [1] => Array
  (
    [domain] => this-domain-isnt-taken.com.au
    [available] => 1
    [costPrice] => 20.95
    [premium] => false
  )
  [2] => Array
  (
    [domain] => testing1234.com
    [available] => 0
  )
)
```

## Example of a failed check

```
[status] => ERR_OCCURED
[errorMessage] => Failed to list domains. Please try again later
```

# Domain Pricing

The Domain Pricing command will return all pricing for the TLDs that are assigned to the reseller's Synergy Wholesale account.

**Command:** getDomainPricing

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

```
Example of successful domain pricing

[status] => OK
[errorMessage] => Pricing has been retrieved
[pricing] => Array
(
  [0] => stdClass Object
  (
    [tld] => me
    [minPeriod] => 1
    [maxPeriod] => 10
    [transfer] => 10.00
    [renew] => 10.00
    [redemption] => 100.00
    [register_1_year] => 10.00
    [register_2_year] => 20.00
    [register_3_year] => 30.00
    [register_4_year] => 40.00
    [register_5_year] => 50.00
    [register_6_year] => 60.00
    [register_7_year] => 70.00
    [register_8_year] => 80.00
    [register_9_year] => 90.00
    [register_10_year] => 100.00
    [sale] => Array
    (
      [transfer] => 5.00
      [renew] => 5.00
      [register_1_year] => 5.00
      [register_2_year] => 15.00
      [register_3_year] => 25.00
      [register_4_year] => 35.00
      [register_5_year] => 45.00
      [register_6_year] => 55.00
      [register_7_year] => 65.00
      [register_8_year] => 75.00
      [register_9_year] => 85.00
      [register_10_year] => 95.00
      [start_sale_date] => 2022-10-01T00:00:00+11:00
      [end_sale_date] => 2022-10-31T11:59:59+11:00
    )
  )
)
```

**Note:** Only pricing periods, and products that are available to the Synergy Wholesale reseller account will be shown. For example, if 'redemption' is not returned in the response then the reseller account does not have access to redemption on that extension. If 'register\_6\_year' (or other fields) are missing then the domain name cannot be registered for 6 years (or the omitted response in other cases).

## Register (.au specific) - Deprecated

The registration of a .AU domain name is specific due to the restrictions in place for the extension. This method is now deprecated, please use domainRegister.

**Command:** domainRegisterAU

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
years	Mandatory	Registration period of domain name
nameServers	Optional	Name servers for the domain name (leave empty to park)
autoRenew	Optional	Enable automatic renewal on the domain name at registration time (provide 'true' to enable). Default behavior is 'false'
registrant_firstname	Mandatory	Registrant First Name
registrant_lastname	Mandatory	Registrant Last Name
registrant_organisation	Optional	Registrant Organisation
registrant_address	Mandatory	Registrant Address (Array data required)
registrant_suburb	Mandatory	Registrant Suburb
registrant_state	Mandatory	Registrant State - Restrictions apply please refer to State List
registrant_country	Mandatory	Registrant Country (2 letter country code) - Refer to Country List
registrant_postcode	Mandatory	Registrant Post code
registrant_phone	Mandatory	Registrant Phone
registrant_fax	Optional	Registrant Fax
registrant_email	Mandatory	Registrant Email Address
<b>Technical Requirements</b> You must also provide the above variables for the technical contact of the domain name registration.		
registrantName	Mandatory	License holder of the domain name
registrantID	Optional	Please refer to the section above of ".AU Domain Registration Information" for explanations of these variables.  Examples of how these variables are used can be viewed in the section "Example - .AU Domain Registration"
registrantIDType	Optional	
eligibilityType	Mandatory	
eligibilityName	Optional	
eligibilityID	Optional	
eligibilityIDType	Optional	
<b>Response</b>		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the registration
<b>Example of successful domain registration</b>		

[errorMessage] => Domain Register AU - Domain Has Been Submitted For Registration

## Register (.uk specific) - Deprecated

The registration of a .UK domain name is specific due to the restrictions in place for the extension. This method is now deprecated, please use domainRegister.

**Command:** domainRegisterUK

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
years	Mandatory	Registration period of domain name
nameServers	Optional	Name servers for the domain name (leave empty to park)
autoRenew	Optional	Enable automatic renewal on the domain name at registration time (provide 'true' to enable). Default behavior is 'false'
registrant_firstname	Mandatory	Registrant First Name
registrant_lastname	Mandatory	Registrant Last Name
registrant_organisation	Optional	Registrant Organisation
registrant_address	Mandatory	Registrant Address (Array data required)
registrant_suburb	Mandatory	Registrant Suburb
registrant_state	Mandatory	Registrant State - Refer to State List
registrant_country	Mandatory	Registrant Country (2 letter country code) - Refer to Country List
registrant_postcode	Mandatory	Registrant Post code
registrant_phone	Mandatory	Registrant Phone
registrant_fax	Optional	Registrant Fax
registrant_email	Mandatory	Registrant Email Address

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the registration

### Example of successful domain registration

```
[status] => OK
[errorMessage] => Domain Register UK Successful - Domain Has Been Submitted For Registration
[costPrice] => 0.00
```

### Example of failed domain registration

```
[status] => ERR_DOMAINREGISTERUK_FAILED
[errorMessage] => Domain Register UK Failed - Domain Already Pending Registration
```

## Register (.us specific) - Deprecated

The registration of a .US domain name is specific due to the restrictions in place for the extension. This method is now deprecated, please use domainRegister.

**Command:** domainRegisterUS

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
years	Mandatory	Registration period of domain name
nameServers	Optional	Name servers for the domain name (leave empty to park)
autoRenew	Optional	Enable automatic renewal on the domain name at registration time (provide 'true' to enable). Default behavior is 'false'.
registrant_firstname	Mandatory	Registrant First Name
registrant_lastname	Mandatory	Registrant Last Name
registrant_organisation	Optional	Registrant Organisation
registrant_address	Mandatory	Registrant Address (Array data required)
registrant_suburb	Mandatory	Registrant Suburb
registrant_state	Mandatory	Registrant State - Refer to State List
registrant_country	Mandatory	Registrant Country (2 letter country code) - Refer to Country List
registrant_postcode	Mandatory	Registrant Post code
registrant_phone	Mandatory	Registrant Phone
registrant_fax	Optional	Registrant Fax
registrant_email	Mandatory	Registrant Email Address
appPurpose	Mandatory	Please refer to section ".US Domain Registration Information"
nexusCategory	Mandatory	

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the registration

### Example of successful domain registration

```
[status] => OK
[errorMessage] => Domain Register US Successful - Domain Has Been Submitted For Registration
[costPrice] => 0.00
```

### Example of failed domain registration

```
[status] => ERR_DOMAINREGISTERUS_FAILED
[errorMessage] => Domain Register US Failed - Domain Already Pending Registration
```

# Register (all other extensions)

The registration of the domain name is specific due to the restrictions in place for the extensions.

**Command:** domainRegister

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be registered
years	Mandatory	Registration period of domain name
nameServers	Optional	Name servers for the domain name (leave empty to park)
idProtect	Optional	Enable ID Protection during registration (provide 'true' to enable)
specialConditionsAgree ( <b>Deprecated</b> )	Optional	For extensions that require specific eligibility confirmation (provide 'true' to agree)
autoRenew	Optional	Enable automatic renewal on the domain name at registration time (provide 'true' to enable). Default behavior is 'false'.
premium	Optional	Boolean value to indicate that the partner accepts the premium pricing terms
costPrice	Optional	The price of the domain registration retrieved from checkDomain. <i>Only required to register premium domain names.</i>
categories	Optional	Array - Categories you wish to assign on transfer complete. Categories that do not exist will be automatically created.
transferLock	Optional	Bool - Apply transfer lock when the domain transfer completes.
eligibility	Optional	A key/value JSON encoded string with the applicable eligibility information. See <i>getDomainEligibilityFields</i>
registrant_firstname	Mandatory	Registrant First Name
registrant_lastname	Mandatory	Registrant Last Name
registrant_organisation	Optional	Registrant Organisation
registrant_address	Mandatory	Registrant Address (Array data required)
registrant_suburb	Mandatory	Registrant Suburb
registrant_state	Mandatory	Registrant State - Refer to State List
registrant_country	Mandatory	Registrant Country (2 letter country code) - Refer to Country List
registrant_postcode	Mandatory	Registrant Post Code
registrant_phone	Mandatory	Registrant Phone
registrant_fax	Optional	Registrant Fax
registrant_email	Mandatory	Registrant Email Address
<b>Billing, Admin and Technical Requirements</b> You may also provide the above variables for the billing, technical and admin contacts of the domain name registration. If a contact set is required but not provided, the contact set for the registrant contact will be applied to each missing contact set..		

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the registration

#### Example of premium domain registration request

```
[resellerID] => 1
[apiKey] => *****
[domainName] => premium.accountants
[years] => 1
[registrant_firstname] => John
[registrant_lastname] => Smith
[registrant_address] => [
  1 Applebee Park Way
]
[registrant_suburb] => Cupertino
[registrant_state] => CA
[registrant_country] => US
[registrant_postcode] => 95014
[registrant_phone] => +1.555555555
[registrant_organisation] => Synergy Wholesale
[idProtect] => true
[autoRenew] => true
[premium] => true
[costPrice] => 151.50
```

#### Example of domain registration request with eligibility

```
[resellerID] => 1
[apiKey] => *****
[domainName] => synergywholesale.com.au
[years] => 1
[registrant_firstname] => John
[registrant_lastname] => Smith
[registrant_address] => [
  1 Applebee Park Way
]
[registrant_suburb] => Cupertino
[registrant_state] => CA
[registrant_country] => US
[registrant_postcode] => 95014
[registrant_phone] => +1.555555555
[registrant_organisation] => Synergy Wholesale
[autoRenew] => true
[eligibility] => {
  "eligibilityType": "Company",
  "registrantName": "SYNERGY WHOLESAL PTY LTD",
  "registrantID": "17165245901",
  "registrantIDType": "ABN"
}
```

#### Example of successful domain registration

```
[status] => OK
[errorMessage] => Domain Register Successful - Domain Has Been Submitted For Registration
[costPrice] => 151.50
```

#### Example of failed domain registration

```
[status] => ERR_DOMAINREGISTER_FAILED
[errorMessage] => Domain Register Failed - Domain Not Available To Register
```

# Is Transferrable

The is domain transferable function will validate if the provided domain password is valid, as well as returning the current expiry date of the domain.

**Command:** isDomainTransferrable

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
authInfo	Mandatory	Domain password/EPP/AuthInfo

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
expiry	String	Domain expiry if available

Example of successful domain transfer
[status] => OK [errorMessage] => Is Domain Transferable Success - Domain Is Able To Be Submitted For Transfer [expiry] => 2023-11-14T00:58:33.0Z

Example of failed domain transfer
[status] => ERR_ISDOMAINTRANSFERRABLE_FAILED [errorMessage] => Is Domain Transferrable Failed - This domain name is already in your account.

# Transfer

The domain transfer function will validate the domain password and if it's available for transfer. If the domain is locked or within a recently renewed period the errorMessage will provide a reason or reasons the domain isn't able to be submitted.

**Note:** Your nameservers will not be updated if the "nameServers" and "dnsConfig" parameters are omitted. Be aware that supplying a dnsConfig value and no nameservers, will automatically apply the nameservers for the appropriate DNS Configuration on transfer completion.

**Command:** transferDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
authInfo	Mandatory	Domain password/EPP/AuthInfo
firstname	Mandatory	First Name
lastname	Mandatory	Last Name
address	Mandatory	Address (Array data required)
suburb	Mandatory	Suburb
state	Mandatory	State - Refer to State List
country	Mandatory	Country (2 letter country code) - Refer to Country List
postcode	Mandatory	Post Code
phone	Mandatory	Phone
email	Mandatory	Email Address
idProtect	Optional	Enable ID Protection during transfer (provide Y to enable)
doRenewal	Optional	.au specific - disable renewal within 90 days (provide 0 to disable)
eligibility	Optional	A key/value JSON encoded string with the applicable eligibility information. See <i>getDomainEligibilityFields</i> . Required for domains that have contact based eligibility i.e. .us, .jobs, .pro
categories	Optional	Array - Categories you wish to assign on transfer complete. Categories that do not exist will be automatically created.
transferLock	Optional	Bool - Apply transfer lock when the domain transfer completes.
nameServers	Optional	Array - Nameservers to change over to when the transfer completes. Note that dnsConfig must be "1" for "Custom Nameservers" for these nameserver to take effect.
dnsConfig	Optional	Refer to the "DNS Configuration Information" section in this document.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the transfer

Example of successful domain transfer
[status] => OK [errorMessage] => Domain Has Been Submitted For Transfer [costPrice] => 0.00

**Example of failed domain transfer**

[status] => ERR\_TRANSFERDOMAIN\_FAILED  
[errorMessage] => Transfer Domain Failed - Domain Status Prohibits Operation

## Transfer (.uk specific) - Deprecated

Domain name transfers for UK domain names are completely different to other extensions. The domain name must first exist in the wholesale before it is assigned to our tag. Once the domain has been submitted you need to request the existing registrar assign it to our tag mentioned below. Our system will automatically detect the inbound transfer to us and complete the process and set the domain live in your account.

This method is now deprecated, please use domainRegister.

**Command:** domainTransferUK

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
authInfo	Optional	Domain name password (internal transfers between Synergy Wholesale partners only)
email	Mandatory	Registrant email used to send the registrant the next steps for transfer

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

### Example of successful domain transfer

```
[status] => OK  
[errorMessage] => Domain Transfer Has Been Submitted Successfully
```

### Example of failed domain transfer

```
[status] => ERR_DOMAINTRANSFERUK_FAILED  
[errorMessage] => Transfer Domain UK Failed - Domain Status Prohibits Operation
```

## Transfer Release (.uk specific)

Domain name transfers for .UK domain names are completely different to other extensions. If you are requested to release/push a domain to another registrar this process must be followed.

**Command:** domainReleaseUK

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
tagName	Mandatory	New UK Tag for the new registrar

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

### Example of successful domain transfer

```
[status] => OK  
[errorMessage] => Domain Release Completed Successfully
```

### Example of failed domain transfer

```
[status] => ERR_DOMAINRELEASEUK_FAILED  
[errorMessage] => Domain Release Failed - Not A Valid .UK Domain Name
```

# Resend Transfer Approval Email

If the registrant of a domain transfer didn't receive the approval email this function will allow you to resend it. We will lookup the registrant email address from the registry in case it has changed prior to sending the email.

**Command:** resendTransferEmail

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
newEmail	String	Email address of the registrant where the email is sent

## Example of successful transfer email resent

```
[newEmail] => you@domainname.com.au  
[status] => OK  
[errorMessage] => Resend Transfer Email Completed
```

## Example of failed transfer email resent

```
[status] => ERR_RESENDTRANSFEREMAIL_FAILED  
[errorMessage] => Resend Transfer Email Failed - Unable To Resend Email
```

## Resubmit Failed Transfer - Deprecated

If the registrant of a domain doesn't accept the transfer within the available time it'll end up in a timeout queue for resubmission. The following command allows for customers to resubmit failed transfers.

**Command:** resubmitFailedTransfer

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
newPassword	Mandatory	Password of the domain name being submitted for transfer

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
newEmail	String	Email address of the registrant where the email is sent
costPrice	String	Amount charged to account for the failed transfer

### Example of successful transfer resubmit

```
[status] => OK  
[errorMessage] => Resubmit Failed Transfer Completed Successfully  
[costPrice] => 0.00
```

### Example of failed transfer resubmit

```
[status] => ERR_RESUBMITFAILEDTRANSFER_FAILED  
[errorMessage] => Resubmit Failed Transfer Failed - Domain Does Not Exist In Valid Stat
```

## Cancel Inbound Transfer

If the domain transfer is not timed out, awaiting refresh, or manual review; it may still be cancelled. Cancelling an inbound transfer will immediately terminate the inbound transfer request with the registry and refund the debited amount to the partner's account. If a transfer is resubmitted post-cancellation the transfer will still take the same amount of time to complete (1 - 5 days, depending on the registry).

**Command:** transferCancel

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
refundAmount	Decimal	The amount refunded for the transfer

Example of successful transfer cancellation
[refundAmount] => 13.95 [status] => OK [errorMessage] => Inbound transfer cancelled

Example of failed transfer cancellation
[status] => ERR_TRANSFERCANCEL_FAILED [errorMessage] => Transfer Cancel Failed - The provided domain name cannot be found

## Approve Outbound Transfer (ack)

The following function allows for pending outbound transfers to be approved (often referred to as 'ack') by the losing registrar allowing for (in most cases) the domain name to be released immediately to the new registrar.

**Command:** transferOutboundApprove

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful pending outbound transfer approval
[status] => OK [errorMessage] => The domain transfer has been approved

Example of failed pending outbound transfer approval
[status] => ERR_TRANSFEROUTBOUNDAPPROVED_FAILED [errorMessage] => Failed to approve transfer at the registry

# Reject Outbound Transfer

The following function allows for pending outbound transfers to be rejected by the losing registrar which causes the pending outbound transfer to be cancelled. **NOTE:** This command does not work with all TLD strings and valid, registry and regulator approved justification for utilisation of this function is required in all cases. Failure to provide a valid and authorized reason will result in immediate suspension of your partner account.

**Command:** transferReject

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
reason	Mandatory	A valid, correct and true reason for the rejection of an outbound domain transfer

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful rejection of an outbound pending transfer

```
[status] => OK  
[errorMessage] => The domain transfer has been rejected successfully
```

## Example of failed rejection of an outbound pending transfer

```
[status] => ERR_TRANSFERREJECT_FAILED  
[errorMessage] => Transfer Reject Failed - <reason for failure>
```

# Renew

To renew a domain name the following variables are required to be sent to the API.

**Command:** renewDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
years	Mandatory	Renewal period in years
premium	Optional	Boolean value required to acknowledge the premium cost price for premium domain renewals
costPrice	Optional	Only required for premium domain renewals

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the renewal

Example of successful renewal
[status] => OK [errorMessage] => Domain Has Been Renewed [costPrice] => 0.00

Example of failed renewal
[status] => ERR_RENEWDOMAIN_FAILED [errorMessage] => Renew Domain Failed - Invalid Number Of Years Passed

# RGP Restore Domain

This will restore a domain name if it is in the Redemption Grace Period (RGP). In order to confirm acceptance of the RGP renewal, the redemption cost price will need to be passed onto this function (can be retrieved via the 'getDomainPricing' command).

**Command:** restoreDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
redemptionPrice	Mandatory	The price that you are charged for the Redemption Grace Period renewal (can be retrieved via the 'getDomainPricing' command) to confirm the RGP renewal
premium	Optional	Boolean value required to acknowledge the premium cost price for premium domain renewals

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the RGP restoration

Example of successful RGP restoration
[status] => OK [errorMessage] => Domain has been restored [costPrice] => 100.00

Example of failed RGP restoration
[status] => ERR_RESTOREDOMAIN_FAILED [errorMessage] => Restore Domain Failed – Unable to restore

## Update Name Servers

The method to update name servers is slightly different to other registrars and their API. We provide built into the update name server function the ability to park, add forwarding or DNS hosting within the one command. The variable of 'dnsConfig' is optional unless you wish to configure parking, forwarding or DNS hosting. Please refer to the section in this domain labeled "DNS Configuration Information".

If you wish to just update name servers, disregard the dnsConfig and only provide an array of name servers to be used.

**Command:** updateNameServers

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
nameServers	Mandatory	An array of name servers must be provided for updating <b>Note:</b> You must provide the name servers as an array
dnsConfig	Optional	Refer to "DNS Configuration Information" section in this document
skipDefaultARecords	Optional (Boolean integer)	When Boolean of '1' is passed, this function will skip the default A record entries being added for any DNS zone being created when setting dnsConfig to Email/Web Forwarding, Parked or DNS Hosting (see "DNS Configuration Information" section in this document for further information) <b>Note:</b> If not specified this will default to '0' (which means records will be added)

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

### Example of successful name server update

```
[status] => OK  
[errorMessage] => Domain name servers have been updated
```

### Example of failed name server update

```
[status] => ERR_DOMAINLOCK_FAILED  
[errorMessage] => Domain Lock Failed - Unable To Determine Current Domain Status
```

# Updating Contacts

The following function allows for the registrant, admin, billing and technical contacts of a domain name to be updated.

**Note:** ID Protection must first be disabled if it is enabled to allow for the contacts to be updated.

**Command:** updateContact

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
registrant_firstname	Mandatory	Registrant First Name
registrant_lastname	Mandatory	Registrant Last Name
registrant_organisation	Optional	Registrant Organisation
registrant_address	Mandatory	Registrant Address (Array data required)
registrant_suburb	Mandatory	Registrant Suburb
registrant_state	Mandatory	Registrant State - Refer to State List
registrant_country	Mandatory	Registrant Country (2 letter country code) - Refer to Country List
registrant_postcode	Mandatory	Registrant Post Code
registrant_phone	Mandatory	Registrant Phone
registrant_fax	Optional	Registrant Fax
registrant_email	Mandatory	Registrant Email Address
appPurpose	Optional	This field is required only when updating contacts on a .US domain name. Please refer to section ".US Domain Registration Information" for more information.
nexusCategory	Optional	
nz_privacy	Optional	.NZ domain names only. This field may be provided as a boolean value (true/false) when updating the IRPO for a .NZ domain name.

## Billing, Admin and Technical Requirements

To update all contact sets to match the registrant, send only the registrant variables in the request, and all other contact sets will be updated to match. To keep existing details on Admin, Technical or Billing contact sets, send the existing variables for these sets in the request.

## Example of successful contact update

```
[status] => OK  
[errorMessage] => Contacts update successfully
```

## Example of failed contact update

```
[status] => ERR_UPDATECONTACTS_FAILED  
[errorMessage] => Contact update failed - Registrant firstname is missing
```

# Resend Registrant Update Email

The following command allows for the gTLD update of registrant (IRTP-C) confirmation emails to be resent.

**Command:** resendRegistrantUpdateEmails

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful registrant update email resend
[status] => OK [errorMessage] => Registrant update emails have been resent to the unconfirmed recipients

Example of failed registrant update email resend
[status] => ERR_RESENDREGISTRANTUPDATEEMAIL_FAILED [errorMessage] => Failed to resend the registrant update emails

# Cancel Pending Registrant Update

The following command allows for the gTLD update of registrant (IRTP-C) process to be cancelled before the new and/or existing Registrant approves.

**Command:** cancelRegistrantUpdate

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful pending registrant update cancellation
[status] => OK [errorMessage] => Registrant update has been cancelled

Example of failed pending registrant update cancellation
[status] => ERR_CANCELREGISTRANTUPDATE_FAILED [errorMessage] => Failed to cancel the registrant update

# List Contacts

Complete contact details can be retrieved from our system to include the admin, tech, billing and registrant contacts.

**Command:** listContacts

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful contacts list - only registrant and tech are returned for a .au domain name; 'nz\_privacy' key is returned for Registrant contact of .NZ domain names only

```
[status] => OK
[errorMessage] => Domain contacts retrieved
[registrant] => Array
(
  [firstname] => Johnny
  [lastname] => Appleseed
  [organisation] =>
  [address1] => 1 Infinite Loop
  [address2] =>
  [address3] =>
  [suburb] => Cupertino
  [state] => CA
  [country] => USA
  [postcode] => 95014
  [phone] => +1.1112346643
  [fax] =>
  [email] => email@address.com
  [nz_privacy] => false
)
[tech] => Array
(
  [firstname] => Johnny
  [lastname] => Appleseed
  [organisation] =>
  [address1] => 1 Infinite Loop
  [address2] =>
  [address3] =>
  [suburb] => Cupertino
  [state] => CA
  [country] => USA
  [postcode] => 95014
  [phone] => +1.1112346643
  [fax] =>
  [email] => email@address.com
)
```

## Example of failed contacts list

```
[status] => ERR_LISTCONTACTS_FAILED
[errorMessage] => List Contacts Failed - Unable to retrieve domain id
```

# List ID Protected Contacts

Retrieve the protected (hidden) admin, tech, billing and registrant contacts set on a domain name that has ID protection enabled.

**Command:** listProtectedContacts

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful protected contacts list
<pre>[status] =&gt; OK [errorMessage] =&gt; Domain contacts retrieved [registrant] =&gt; Array (   [firstname] =&gt; Johnny   [lastname] =&gt; Appleseed   [company] =&gt;   [address1] =&gt; 1 Infinite Loop   [address2] =&gt;   [address3] =&gt;   [suburb] =&gt; Cupertino   [state] =&gt; CA   [country] =&gt; USA   [postcode] =&gt; 95014   [phone] =&gt; +1.1112346643   [fax] =&gt;   [email] =&gt; email@address.com ) [tech] =&gt; Array (   [firstname] =&gt; Johnny   [lastname] =&gt; Appleseed   [company] =&gt;   [address1] =&gt; 1 Infinite Loop   [address2] =&gt;   [address3] =&gt;   [suburb] =&gt; Cupertino   [state] =&gt; CA   [country] =&gt; USA   [postcode] =&gt; 95014   [phone] =&gt; +1.1112346643   [fax] =&gt;   [email] =&gt; email@address.com )</pre>

Example of failed protected contacts list
<pre>[status] =&gt; ERR_LISTPROTECTEDCONTACTS_FAILED [errorMessage] =&gt; ID Protection is not enabled on this domain. Use "listContacts" to get the domain contacts.</pre>

# Get Raw Contacts

Complete contact details can be retrieved from the domain registry including the admin, tech, billing and registrant contacts where applicable based on the TLD of the queried domain.

**Command:** rawDomainContacts

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
authInfo	Mandatory	Domain name Auth info

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

**Example of successful contacts list - only registrant and tech are returned for a .au domain name; 'nz\_privacy' key is returned for Registrant contact of .NZ domain names only**

```
[status] => OK
[errorMessage] => Contact details successfully retrieved from registry.
[registrant] => Array
(
    [firstname] => Johnny
    [lastname] => Appleseed
    [company] =>
    [address1] => 1 Infinite Loop
    [address2] =>
    [address3] =>
    [suburb] => Cupertino
    [state] => CA
    [country] => USA
    [postcode] => 95014
    [phone] => +1.1112346643
    [fax] =>
    [email] => email@address.com
    [nz_privacy] => false
)

[tech] => Array
(
    [firstname] => Johnny
    [lastname] => Appleseed
    [company] =>
    [address1] => 1 Infinite Loop
    [address2] =>
    [address3] =>
    [suburb] => Cupertino
    [state] => CA
    [country] => USA
    [postcode] => 95014
    [phone] => +1.1112346643
    [fax] =>
    [email] => email@address.com
)
```

**Example of failed contacts list**

```
[status] => ERR_RAW_DOMAIN_CONTACTS
[errorMessage] => Incorrect Domain AuthInfo
```

# Add Registry Host

Add a new registry host to be used as a name server for a domain name. You must have added the registry host prior to trying to use it as a name server on a domain name.

**Command:** addHost

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
host	Mandatory	Hostname of the registry host you are adding (excluding domain)
ipAddress	Mandatory	IPv4 or IPv6 Address - must be provided as array

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful adding of registry host

```
[status] => OK  
[errorMessage] => Add Host Completed Successfully
```

## Example of failed adding of registry host

```
[status] => ERR_ADDHOST_FAILED  
[errorMessage] => Add Host Failed
```

# Delete Registry Host

Add a new IP address to a registry host you must ensure the host already exists in the registry.

**Command:** deleteHost

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
host	Mandatory	Hostname of the registry host you are deleting

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful deleting of the registry host

```
[status] => OK  
[errorMessage] => Delete Host Successfully Completed
```

## Example of failed deleting the registry host

```
[status] => ERR_DELETEHOST_FAILED  
[errorMessage] => Delete Host Failed - Host Does Not Exist
```

# Add Registry Host IP Address

Add a new IP address to a registry host you must ensure the host already exists in the registry.

**Command:** addHostIP

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
host	Mandatory	Hostname of the registry host you are adding the IP address to
ipAddress	Mandatory	IPv4 or IPv6 Address - must be provided as array

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful adding of registry host IP address
[status] => OK [errorMessage] => Add Host IP Completed Successfully

Example of failed adding of registry host IP address
[status] => ERR_ADDHOSTIP_FAILED [errorMessage] => Add Host IP Address Failed

# Delete Registry Host IP Address

Delete an IP address to a registry host you must ensure the host already exists in the registry.

**Command:** deleteHostIP

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
host	Mandatory	Hostname of the registry host you are delete the IP address from
ipAddress	Mandatory	IPv4 or IPv6 Address - must be provided as array

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful deleting of registry host IP address
[status] => OK [errorMessage] => Delete Host IP Completed Successfully

Example of failed deleting of registry host IP address
[status] => ERR_DELETEHOSTIP_FAILED [errorMessage] => Delete Host IP Address Failed

# Registry Host Information

Information relating to the specified registry host will be returned using this command.

**Command:** listHost

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
host	Mandatory	Hostname of registry host (excluding domain name)

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
host	String	Hostname you are looking up the information
domainName	String	Domain name of the registry host
ipAddress	Array	IP address(es) associated with the registry host

## Example of successful registry host lookup

```
[status] => OK
[errorMessage] => Host Listing Completed Successfully
[host] => ns1
[domainName] => testdomain.id.au
[ipAddress] => Array
(
  [0] => 110.232.142.138
  [1] => 2400:b800:2:1:0:ffff:4e92:9f3a
)
```

## Example of failed registry host lookup

```
[status] => ERR_LISTHOST_FAILED
[reason] => Host List Failed - No Host Exists
```

# List All Registry Hosts

This command will return the registry hosts assigned to the specified domain name.

**Command:** listAllHosts

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
hosts	Array	Hosts that exist for the specified domain name
hostName	String	Hostname that exists at the registry
ip	Array	IP address(es) assigned to the registry host

## Example of successful registry hosts lookup

```
[status] => OK
[errorMessage] => List All Hosts Completed Successfully
[hosts] => Array
(
  [0] => Array
  (
    [hostName] => ns1.testdomain.net.au
    [ip] => Array
    (
      [0] => 112.140.176.10
    )
  )
  [1] => Array
  (
    [hostName] => ns2.testdomain.net.au
    [ip] => Array
    (
      [0] => 112.140.180.10
    )
  )
)
```

## Example of failed registry hosts lookup

```
[status] => ERR_LISTALLHOSTS_FAILED
[errorMessage] => List All Hosts Failed - No Host Records Present
```

# Transfer Lock

When you enable the domain name transfer lock it will prevent any other registrar being able to transfer the domain name away from your account. If you or a customer wishes to transfer the domain away the transfer lock must be disabled.

**Command:** lockDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful transfer lock

```
[status] => OK  
[errorMessage] => Domain Has Been Locked
```

## Example of failed transfer lock

```
[status] => ERR_DOMAINLOCK_FAILED  
[errorMessage] => Domain Lock Failed - Unable To Determine Current Domain Status
```

# Transfer Unlock

Domain names that have been unlocked are available for transfer to any other registrar.

**Command:** unlockDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful transfer unlock
[status] => OK [errorMessage] => Domain Has Been unlocked

Example of failed transfer unlock
[status] => ERR_DOMAINUNLOCK_FAILED [errorMessage] => Domain Unlock Failed - Unable To Determine Current Domain Status

# Domain Renew Required Check

This command will check whether a domain name renewal is required upon transfer request

**Command:** domainRenewRequired

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
authInfo	Mandatory	The domain name transfer password

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful transfer unlock
[status] => OK_NO_RENEWAL [errorMessage] => Domain Does Not Require Renewal [domain_expiry] => 2016-07-26 10:23:00

Example of failed transfer unlock
[status] => ERR_DOMAINRENEWREQUIRED_FAILED [errorMessage] => Domain Renew Required Failed - Unable To Check For Renewal Status

# Max Years Domain Can Be Renewed For

This command will return the maximum number of years a particular domain name can be renewed for. This takes into account the time remaining on the domain name, plus the maximum period supported by the registry to return a specific result for a queried domain name.

**Command:** maxYearsCanRenewFor

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
maxYears	String	Number of full years the domain name can be renewed for

## Example of successful transfer unlock

```
[status] => OK  
[errorMessage] => Max Years Can Renew For Completed Successfully  
[maxYears] => 2
```

## Example of failed transfer unlock

```
[status] => ERR_MAXYEARS_CANRENEWFOR_FAILED  
[errorMessage] => Max Years Can Renew For Failed - Domain not eligible for renewal
```

# List Available Domain Extensions

This command will return the domain name extensions available to a reseller, including the category that they are categorised in. There are a total of 16 categories and a domain name may have multiple categories assigned. The categories are: Most Popular, Australian, International, New Zealand, Personal, Business, Commerce, Education, Entertainment, Arts & Media, Health, Finance, Professional, Industry, Sports and Technology. If you feel a domain extension should belong to a particular category or have any feedback on the category allocations please contact the support team via ticket.

**Command:** listAvailableDomainExtensions

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
extensions	Array	Array of extensions available to reseller, including assigned category

Example of successful listAvailableDomainExtensions
<pre>[status] =&gt; OK [errorMessage] =&gt; Extension list retrieval successful [extensions] =&gt; [   [tid] =&gt; com   [tidType] =&gt; gTLD   [categories] =&gt; [     Most Popular     Business   ] ]</pre>

Example of failed listAvailableDomainExtensions
<pre>[status] =&gt; ERR_RESELLER_NOT_AUTHORISED [errorMessage] =&gt; List Extensions Failed - It would appear you do not have access to any domain name extensions.</pre>

## .XXX Membership Details

.XXX domain names require the membership process to be completed with the ICM registry prior to the domain name resolving. An email is sent to the registrant email address provided at the time of registration. This function allows for the .XXX membership details to be provided.

**Command:** updateXXXMembershipDetails

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
membershipToken	Mandatory	.XXX Membership token provided by the confirmation process

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

### Example of successful XXX Membership details update

```
[status] => OK  
[errorMessage] => Update XXX Membership Details Successfully Completed
```

### Example of failed XXX Membership details update

```
[status] => ERR_UPDATEXXXMEMBERSHIPDETAILS_FAILED  
[errorMessage] => Update XXX Membership Details Failed - Unable To Update Membership ID
```

# Retrieve .US Nexus Data

.US domain names require eligibility data for the purpose of registration. This function will allow you to view the nexus data for the required domain name.

**Command:** getUSNexusData

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
nexusCategory	String	Category used for the .US domain name
nexusApplication	String	Application of the .US domain name

## Example of successful get nexus data

```
[status] => ok  
[errorMessage] => Get US Nexus Data Successfully Completed  
[nexusCategory] => C31  
[nexusApplication] => P2
```

## Example of failed nexus data

```
[status] => ERR_GETUSNEXUSDATA  
[errorMessage] => Unable to retrieve nexus data for supplied domain
```

# .AU Change of Registrant Request

The Change of Registrant for .AU domain names requires manual verification. This function will begin the initial process of sending an email to the registrant email address. Details contained in the email will outline the steps required to complete the Change of Registrant process.

**Important:** The existing registrant must still be valid and eligible for the domain name. This process will confirm the existing ABN or ACN is still Active/Registered prior to allowing the process to continue.

**Command:** initiateAUCOR

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
years	Optional	Number of years to re-register the domain name for

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
costPrice	String	Amount charged to account for the Change of Registrant

## Example of successful Change of Registrant Request

```
[status] => OK  
[errorMessage] => Initiate AU COR Success - COR Has Been Initiated  
[costPrice] => 1.23
```

## Example of failed Change of Registrant Request

```
[status] => ERR_INITIATEAUCOR_FAILED  
[errorMessage] => Initiate AU COR Failed - Unable To Determine If Reseller Owns Domain
```

# Cancel Change of Registrant Request

This function will attempt to cancel an in progress change of registrant request, as long as the request is in a state that may be cancelled.

A change of registrant request may generally be cancelled if the request has not progressed to a 'final' stage whereby the request is considered complete. Note that this function will also return an error if there are no pending change of registrant requests for the given domain name.

**Command:** cancelChangeOfRegistrant

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful Change of Registrant Request Cancellation

[status] => OK  
[errorMessage] => Change of registrant request cancelled successfully.

## Example of failed Change of Registrant Request Cancellation

[status] => ERR\_CANCELOR\_FAILED  
[errorMessage] => The change of registrant request cannot be cancelled at this time.

# Add DNSSEC Record

Outlined below is the necessary requirements for adding a DNSSEC record at the registry for the domain name provided. Certain variables require an identifier to be provided which can be found in the 'DNSSEC Information' table to explain each requirement.

**Tip:** You can use the extension options function to see which domain name extensions support DNSSEC.

**Command:** DNSSECAddDS

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
algorithm	Mandatory	Refer to the section 'DNSSEC Information' for the requirements of this variable
digestType	Mandatory	Refer to the section 'DNSSEC Information' for the requirements of this variable
digest	Mandatory	The generated hexadecimal binary value related to the digest type
keyTag	Mandatory	Your chosen key to assign for the value of the DNSSEC record

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
UUID	String	Returned ID of a successful record being added

## Example of successful adding of a DNSSEC Record

```
[status] => OK  
[UUID] => 3969020
```

## Example of failed adding of DNSSEC Record

```
[status] => ERR_DNSSECDSADD_FAILED  
[errorMessage] => DNSSEC DS Add Failed - Re-check provided details
```

## Example of failed adding of DNSSEC Record

```
[status] => ERR_DNSSECDSADD_WARNING  
[errorMessage] => DNSSEC DS Add Failed - DNSSEC DS Data already exists
```

# Remove DNSSEC Record

The function detailed below allows for the DNSSEC record to be deleted by using the UUID which is able to be obtained from the DNSSECListDS function or stored from adding a record using the DNSSECAAddDS function.

**Command:** DNSSECRemovedDS

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
UUID	Mandatory	Obtained from viewing the record using DNSSECListDS or after creation

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful removal of a DNSSEC record

[status] => OK

## Example of failed removal of DNSSEC record

[status] => ERR\_ERR\_DNSSECDSADD\_FAILED\_FAILED  
[errorMessage] => DNSSEC DS Remove Failed - DNSSEC DS Data does not exist

# List DNSSEC Entries

Retrieval of any DNSSEC records associated with a domain name can be obtained using the function outlined below.

**Command:** DNSSECListDS

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful listing of DNSSEC records
<pre>[status] =&gt; OK [errorMessage] =&gt; List All Hosts Completed Successfully [DSData] =&gt; Array (   [174] =&gt; Array   (     [keyTag] =&gt; 5214     [algorithm] =&gt; 5     [digest] =&gt; 4761674BFF957211D129B0DFE9410AF753559D4B     [digestType] =&gt; 1   ) )</pre>

Example of failed listing of DNSSEC records
<pre>[status] =&gt; ERR_DNSSECDSLIST_FAILED [errorMessage] =&gt; DNSSEC DS List Failed - Registry does not support DNSSEC</pre>

## DNSSEC Information

The information listed below relates to the required ID to be used for the DNSSECAddDS function and when retrieving a list of existing DNSSEC records from the API using DNSSECListDS.

Algorithm	
ID	Explanation
1	RSA/MD5
2	Diffie-Hellman
3	DSA/SHA-1
5	RSA/SHA-1
6	DSA-NSEC3-SHA1
7	RSASHA1-NSEC3-SHA
8	RSA/SHA-256
10	RSA/SHA-512
12	GOST R 34.10-2001
13	ECDSA Curve P-256 with SHA-256
14	ECDSA Curve P-384 with SHA-384

Digest Type	
ID	Explanation
1	SHA-1
2	SHA-256
3	GOST R 34.11-94
4	SHA-384

# Extension Options

There are a number of different options and restrictions for certain domain name extensions. This function is provided to assist our customers using the API to determine which domain name extension has options available for management.

**Command:** getDomainExtensionOptions

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
tld	Mandatory	Extension of the domain name being requested

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
canRenew	String	Determines if a domain name can be renewed
canRenewWithin	String	Number of days a domain can be renewed within
cannotRenewAfter	String	Number of days by which a domain cannot be renewed after expiry
cannotRestoreAfter	String	Number of days by which a domain cannot be renewed after renewal limit
deletesAfter	String	Number of days by which a domain cannot be renewed after restore limit
canRestore	String	Determines if a domain name can be restored
isIPv4Capable	String	Supports IPv4 Child Name Servers
isIPv6Capable	String	Supports IPv6 Child Name Servers
isIDProtectCapable	String	Ability to use ID Protection on requested extension
transferLock	String	Ability to use Domain Lock on requested extension
isHostsCapable	String	Supports Child Name Servers
minYears	String	Minimum years for registration or renewal
maxYears	String	Maximum years for registrations or renewal
DNSSECAvailable	String	Availability of DNSSEC for the extension requested
passwordRequirements	Array	Minimum requirements for the domain name transfer (EPP) password (some requirements set by registry are not mentioned here, such as 'no dictionary words')
availableContacts	Array	Returns the available contact sets for the extension
whoisVerification	Boolean (true/false)	Whether WHOIS verification is required on the extension

## Example of successful extension options for .com.au domain

```
[status] => OK
[errorMessage] => Domain Extension Options Completed Successfully
[canRenew] => yes
[canRenewWithin] => 90
[cannotRenewAfter] => 30
[isIPv4Capable] => yes
[isIPv6Capable] => yes
[isIDProtectCapable] => no
[transferLock] => no
[isHostsCapable] => yes
[minYears] => 2
[maxYears] => 2
[DNSSECAvailable] => 1
```

```
[passwordRequirements] => Array
(
  [minimumLength] => 6
  [maximumLength] => 16
  [complexity] => '/[a-zA-Z0-9()\[\]!@$^*_\-\+=~*]{6,16}'
)
[availableContacts] => Array
(
  [0] => registrant
  [1] => tech
)
[whoisVerification] => false
```

#### Example of failed extension options

```
[status] => ERR_NO_TLD_OPTIONS
[errorMessage] => Unable to locate any tld options for the specified tld
```

# Determine Domain Is Renewable

This function is provided to advise the maximum years and if a domain name is able to be renewed at the current time.

**Command:** canRenewDomain

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
yearsCanRenewFor	String	Number of years the domain is able to be renewed

## Example of successful Domain Renewable

```
[status] => OK_RENEW  
[errorMessage] => Domain Renewal Can Be Processed  
[yearsCanRenewFor] => X
```

## Example of failed Domain Renewable

```
[status] => OK_NO_RENEWAL  
[errorMessage] => Domain Renewal Cannot Be Processed - Already Registered For Maximum Period Available
```

# Transferred Away

The following function is available to customers to obtain a list of domain names that have transferred away.

**Command:** getTransferredAwayDomains

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
startDate	Optional	Exclude outbound transfers before specified date and time (2017-07-13 13:00 or 2017-07-13)
endDate	Optional	Exclude outbound transfers after specified date and time (2017-07-13 13:00 or 2017-07-13)

Response		
Variable	Data Type	Description
domain_id	String	The unique identifier for the outbound transfer
domainname	String	The domain name of the outbound transfer
registrar	String	The name of the gaining registrar
reseller	String	The name of the reseller (if the domain name was transferred internally)
timestamp	String	Timestamp of when the domain name was transferred away
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful list of transferred away domains

```
[domains] => Array
(
  [0] => stdClass Object
  (
    [domainid] => 1799241123
    [domainname] => mynewdomainisapotato.com
    [registrar] => Synergy Wholesale
    [reseller] => VentralP Australia
    [timestamp] => 2015-05-12 12:59:58
    [status] => Transferred Away
  )
  [1] => stdClass Object
  (
    [domainid] => 1799241124
    [domainname] => hellotharsynergy.com
    [registrar] => Synergy Wholesale
    [reseller] => Zuver
    [timestamp] => 2017-07-13 13:00:31
    [status] => Transferred Away
  )
  [3] => stdClass Object
  (
    [domainid] => 1799241125
    [domainname] => whydidyoutransferaway.com
    [registrar] => AUSSIEHQ
    [reseller] =>
    [timestamp] => 2011-07-25 05:40:38
    [status] => Transferred Away
  )
)
```

## Example of failed transferred away domain

```
[status] => ERR_LOGIN_FAILED
```

[errorMessage] => Unable to login to wholesale system

## Resend ICANN Registrant Verification Email

The following function will let you resend the ICANN Registrant Verification email, which has a link to approve the changes to the Registrant contact data. Only changes to the registrant first name, last name and email address will cause the verification email to be sent.

**Note:** If ID protection is enabled on a domain, the ICANN Registrant Verification email will be sent to the ORIGINAL email address and not the ID Protection email address displayed on WHOIS (eg: synergywholesalecom@obscure.me).

**Command:** resendVerificationEmail

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name for the email to be resent

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

### Example of successful resent verification email

[status] => OK  
[errorMessage] => Resend Verification Email Success - Validation Email Has Been Resent

### Example of failed verification email (details already valid)

[status] => ERR\_RESENDVERIFICATIONEMAIL\_FAILED  
[errorMessage] => Resend Verification Email Failed - Domain Contact Details Already Verified

# Add DNS Zone

Use this function to add a new DNS zone into our cluster.

Only domain names which are within your wholesale account can be configured for DNS hosting or URL/Email forwarding. Third party domain names cannot be added.

**Command:** addDNSZone

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful new DNS zone
[status] => OK [errorMessage] => Domain was successfully created in DNS

Example of failed new DNS zone
[status] => ERR_DOMAIN_EXISTS [errorMessage] => Unable to add the domain as it already exists in DNS

# Delete DNS Zone

Use this function to add a new DNS zone into our cluster.

**Command:** deleteDNSZone

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful deletion DNS zone

```
[status] => OK  
[errorMessage] => Domain zone was successfully deleted from DN
```

## Example of failed deletion DNS zone

```
[status] => ERR_DOMAIN_EXISTS  
[errorMessage] => Unable to delete the domain from DNS
```

# Add DNS Record

Add a new DNS record to a domain zone that already exists.

**Command:** addDNSRecord

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
recordName	Mandatory	Hostname of the record
recordType	Mandatory	Record type (MX, A, AAAA, CNAME, TXT)
recordContent	Mandatory	Content of the record
recordTTL	Mandatory	Time to Live of the record
recordPrio	MX Required	Priority of a MX record is required

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful new DNS record
[status] => OK [errorMessage] => Domain Record Was Added Successfully

Example of failed new DNS record
[status] => ERR_VAR_EMPTY [errorMessage] => One of the required variables are missing or empty

# Update DNS Record

Update an existing DNS record to a domain zone that already exists.

**Command:** updateDNSRecord

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
recordID	Mandatory	ID for the record being updated
recordName	Mandatory	Hostname of the record
recordType	Mandatory	Record type (MX, A, AAAA, CNAME, TXT)
recordContent	Mandatory	Content of the record
recordTTL	Mandatory	Time to Live of the record
recordPrio	MX Required	Priority of a MX record is required

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful Update of DNS record

```
[status] => OK  
[errorMessage] => Domain Record Was Updated Successfully
```

## Example of failed to update DNS record

```
[status] => ERR_VAR_EMPTY  
[errorMessage] => One of the required variables are missing or empty  
Or  
[errorMessage] => The DNS hostname '@.example' is not valid.
```

# Delete DNS Record

Delete a DNS record to a domain zone that already exists.

**Command:** deleteDNSRecord

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
recordID	Mandatory	ID of the entry you wish to delete (obtained from List DNS Zone)

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful delete DNS record
[status] => OK [errorMessage] => Domain Record Was Deleted Successfully

Example of failed delete DNS record
[status] => ERR_VAR_EMPTY [errorMessage] => One of the required variables are missing or empty

# Get DNS Record

Retrieve details for a DNS record that already exists.

**Command:** getDNSRecord

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
recordID	Mandatory	ID of the entry you wish to get (obtained from List DNS Zone)

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
domainName	String	Domain name actioned
recordType	String	Record type (eg, MX, TXT, CNAME, etc)
recordContent	String	Set content for the record being actioned
recordID	String	ID for the record
recordTTL	String	Time to Live of the record
recordPrio	String	Priority of the record (if applicable)
errorMessage	String	Text message of the failed command

Example of successful Get DNS record
[status] => OK [errorMessage] => DNS entry successfully retrieved

Example of failed Get DNS record
[status] => ERR_VAR_EMPTY [errorMessage] => One of the required variables are missing or empty

# List DNS Zone

This function will return all records for the domain name supplied during lookup.

**Command:** listDNSZone

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
records	Array	Array of all domain records associated with the domain name

Example of successful DNS zone retrieval
<pre>[hostName] =&gt; exampledomainname.com.au [type] =&gt; NS [content] =&gt; ns1.nameserver.net.au [ttl] =&gt; 86400 [prio] =&gt; [id] =&gt; b17cfe936f7bb5c2c3c14fe07d8c579881144cbdd5bf959db127d2f28fd9f49</pre>

Example of failed DNS zone retrieval
<pre>[status] =&gt; ERR_DBLOOKUP_FAILED [errorMessage] =&gt; Unable to list all dns entries for the specified zone</pre>

**Note:** The ID that is returned for each DNS record is calculated based on the hostHost, type and content, An example of this is provided: DNS Record ID Calculation.

# Add Email Forward

To add a email forward please ensure you use the required variables mentioned below.

**Command:** addMailForward

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
source	Mandatory	Hostname which you'd like to setup as a email forward
destination	Mandatory	Destination where you'd like to redirect the email forward

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful email forward creation

```
[status] => OK  
[destination] => Domain Mail Forward Created Successfully
```

## Example of failed email forward creation

```
[status] => ERR_DBUPDATE_FAILED  
[errorMessage] => URL Forwarding Creation Failed
```

# Delete Email Forward

Use this function to delete the supplied email forwarder.

**Command:** deleteMailForward

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
forwardID	Mandatory	ID of the record you wish to delete

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful deletion of email forward
[status] => OK [errorMessage] => Domain Mail Forward Deleted

Example of failed deletion of email forward
[status] => ERR_RESELLER_NOT_AUTHORISED [errorMessage] => Reseller is not authorized to perform any functions on this domain

# List Email Forwards

Lookup the list of configured email forwarding entries for the supplied domain name.

**Command:** listMailForwards

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
forwards	Array	Array containing the details of the email forwards (see example)

## Example of successful email forwards retrieval

```
[id] => 157  
[source] => sales@testdomain.com.au  
[destination] => sales@anotherdomain.com.au
```

## Example of failed email forwards retrieval

```
[status] => ERR_RESELLER_NOT_AUTHORISED  
[errorMessage] => Reseller is not authorized to perform any functions on this domain
```

# Add Simple URL Forward

To add a URL forward please ensure you use the required variables mentioned below.

**Command:** addSimpleURLForward

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
hostName	Optional	Hostname which you'd like to setup as a URL forward
destination	Mandatory	Destination where you'd like to redirect the hostname
type	Mandatory	Refer to section 'DNS Configuration Information - Redirect Types'
redirectStructure	Optional	Redirect the URL structure with the URL forward
pageTitle	Optional	Page title to be visible when viewing the hostname
metaDescription	Optional	Meta description to be added to the URL forward source
metaKeywords	Optional	Meta keywords to be added to the URL forward source
refreshSeconds	Optional	When using HTML Redirect refresh in X seconds
redirectMessage	Optional	When using HTML Redirect display this message prior to redirection

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful URL forward creation

```
[status] => OK  
[destination] => URL Forwarding Created Successfully
```

## Example of failed URL forward creation

```
[status] => ERR_DBUPDATE_FAILED  
[errorMessage] => URL Forwarding Creation Failed
```

# Delete Simple URL Forward

Use this function to delete the supplied email forwarder.

**Command:** deleteSimpleURLForward

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
recordID	Mandatory	ID of the record you wish to delete

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful deletion of URL forward
[status] => OK [errorMessage] => Simple URL Forward Deleted

Example of failed deletion of URL forward
[status] => ERR_RESELLER_NOT_AUTHORISED [errorMessage] => Reseller is not authorized to perform any functions on this domain

# List Simple URL Forwards

Lookup the list of configured URL forwarding entries for the supplied domain name.

**Command:** getSimpleURLForwards

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
records	Array	Array containing the details of the URL forwards (see example)

Example of successful URL forwards retrieval
<pre>[hostname] =&gt; www.exampledomain.com.au [recordID] =&gt; [destination] =&gt; <a href="http://www.testdomain.com.au">http://www.testdomain.com.au</a> [redirectType] =&gt; P [pageTitle] =&gt; [metaKeywords] =&gt; [metaDescription] =&gt; [refreshSeconds] =&gt; 0 [h1Title] =&gt; [redirectMessage] =&gt; [redirectStructure] =&gt; 0</pre>

Example of failed URL forwards retrieval
<pre>[status] =&gt; ERR_RESELLER_NOT_AUTHORISED [errorMessage] =&gt; Reseller is not authorized to perform any functions on this domain</pre>

# Enable ID Privacy Protection

Enable ID Privacy Protection to hide the contact details of a domain name to be replaced with a generic forwarding alias and postal information.

**Command:** enableIDProtection

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful ID Privacy Protection enabled

```
[status] => OK  
[errorMessage] => ID Protection Enabled Successfully
```

## Example of failed enabling of ID Privacy Protection

```
[status] => ERR_RESELLER_NOT_AUTHORISED  
[errorMessage] => Reseller is not authorized to perform any functions on this domain
```

# Disable ID Privacy Protection

Disabling ID Privacy Protection after it has been enabled will restore the previous contact information for all contacts required for the domain name extension.

**Command:** disableIDProtection

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful ID Privacy Protection disabled

```
[status] => OK  
[errorMessage] => ID Protection Has Now Been Disabled
```

## Example of failed disabling of ID Privacy Protection

```
[status] => ERR_RESELLER_NOT_AUTHORISED  
[errorMessage] => Reseller is not authorized to perform any functions on this domain
```

# Enable Automatic Renewal

Enabling automatic domain name renewal will ensure you don't have to worry about manually renewing the domain name if you miss the expiry date.

**Note:** Automatic renewal cannot be enabled on premium domains.

**Command:** enableAutoRenewal

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful automatic renewal enabled

```
[status] => OK  
[errorMessage] => Auto Renewal Has Been Activated Successfully
```

## Example of failed enabling of automatic renewal

```
[status] => ERR_RESELLER_NOT_AUTHORIZED  
[errorMessage] => Reseller is not authorized to perform any functions on this domain
```

# Disable Automatic Renewal

Disabling automatic domain renewal will no longer have the domain name renewed without interaction. The domain name must be renewed manually or risk the domain name being deleted from the registry.

**Command:** disableAutoRenewal

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful automatic renewal disabled

```
[status] => OK  
[errorMessage] => Auto Renewal Has Been Deactivated Successfully
```

## Example of failed enabling of automatic disabled

```
[status] => ERR_RESELLER_NOT_AUTHORISED  
[errorMessage] => Reseller is not authorized to perform any functions on this domain
```

# Lookup ABN/ACN/RBN Information

This function will allow you to retrieve information relating to the ABN/ACN/RBN provided to ensure you provide the correct registrant details to the registration of a .au domain name.

**Command:** businessCheckRegistration

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
registrationNumber	Mandatory	ABN/ACN/RBN number to be looked up at the relevant online search
registrationState	Optional	Registration state is required for looking up state based business numbers

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful ABN number lookup

```
[status] => OK
[errorMessage] => Business Check Registration Successfully Completed
[registrationNumber] => 11 123 123 123
[entityStatus] => Active
[asicNumber] =>
[entityName] =>
[entityTypeCode] =>
[tradingName] =>
[legalName] => SMITH, JOHN APPLE
[organisationType] => Individual/Sole Trader
[state] => VIC
[postcode] => 3000
```

## Example of failed ABN/ACN/RBN number lookup

```
[status] => ERR_BUSINESSCHECKREGISTRATION_FAILED
[errorMessage] => Business Check Registration Failed - Unable to locate business details
```

Note: The entityTypeCode will map to <https://www.abr.gov.au/government-agencies/accessing-abr-data/abr-data-dictionary/entity-mapping-file>

# Generate .au Eligibility from Business Number

This function will allow you to retrieve eligibility details for a .au domain from the business number (ABN/ACN/RBN) and return it in the format required to pass through to the domainRegister command's eligibility field. It will automatically determine the type of business number retrieve the information from the appropriate business registry.

**Command:** generateAuEligibility

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
registrationIdentifier	Mandatory	ABN/ACN/RBN number to be looked up at the relevant online search

## RBN format for registrationIdentifier

To ensure you receive accurate results, when providing an RBN you must specify the state before the registration like so "VIC B123456U". Otherwise, the API will lookup the RBN against each state until it finds a match.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
eligibility	Object	
eligibilityType	String	The eligibility type determined from the entity
registrantName	String	The name of the registrant according to the registration. <i>This field is required for all .AU registrations and will ONLY be auto filled if the registrationIdentifier is an ABN or ACN.</i>
registrantIDType	String	One of the following values; ABN, ACN, OTHER
registrantID	String	The ABN or ACN of the registrant (if applicable)
eligibilityIDType	String	The type of registrationIdentifier provided (when not ABN or ACN)
eligibilityID	String	The RBN or TRADEMARK number <i>Not returned if provided registrationIdentifier is an ACN or ABN.</i>
eligibilityName	String	The name of the entity. <i>Not returned if provided registrationIdentifier is an ACN or ABN.</i>

## Example of successful ABN number lookup

```
[status] => OK
[errorMessage] => Generate AU Eligibility Successfully Completed
[eligibility] => stdClass Object (
  [eligibilityType] => Company
  [registrantName] => SYNERGY WHOLESALE PTY LTD
  [registrantID] => 17165245901
  [registrantIDType] => ABN
)
```

## Example of successful RBN number lookup

```
[status] => OK
[errorMessage] => Generate AU Eligibility Successfully Completed
[eligibility] => stdClass Object (
  [registrantIDType] => OTHER
  [eligibilityIDType] => VIC BN
  [eligibilityID] => B2164382Y
  [eligibilityType] => Registered Business
  [eligibilityName] => 1300 LOCKSMITH
)
```

**Example of failed ABN/ACN/RBN number lookup**

[status] => ERR\_GENERATEAUELIGIBILITY\_FAILED  
[errorMessage] => Generate AU Eligibility Failed - Business Registration not found

[status] => ERR\_GENERATEAUELIGIBILITY\_FAILED  
[errorMessage] => Generate AU Eligibility Failed - ABN is not active.

# Get Eligibility Fields

This function will allow you to retrieve eligibility fields for the provided extension.

**Command:** getDomainEligibilityFields

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
extension	Mandatory	The domain extension
type	Optional	Provides the values "DOMAIN", or "CONTACT" to get the eligibility data for a specific object. If omitted, all types are returned.  <b>Note:</b> Confirmations do not have a "type" and will only be returned if this parameter is omitted.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
fields	Array	Array containing fields
name	String	The name of the field. This is the key value of the eligibility value passed to domainRegister.
title	String	The user friendly field name
description	String	A description of the field name. If applicable, usually contains user-friendly instructions to instruct the users on how to use the fields.
required	Integer	Boolean value, specifies whether the field is mandatory
policySummary	String	A summary of the registration policy (only returned if applicable)
type	String	One of the following values "checkbox", "text", "select"
options	Array	Contains a key/value array of options ( <i>only for fields that have a "type" of "select"</i> )
value	String	The value that would make up your key/value eligibility parameter for domainRegister
description	String	The user-friendly label for the option

## Note regarding HTML content

Please note that the title, description, and policySummary fields may contain HTML.

#### Example of successful eligibility field retrieval for .com.au

```
[status] => OK_ELIGIBILITY
[errorMessage] => This extension has eligibility criteria.
[fields] => Array
(
    [0] => stdClass Object
        (
            [name] => registrantName
            [title] => Registrant Name
            [description] =>
            [required] => 1
            [type] => text
        )
    [1] => stdClass Object
        (
            [name] => registrantID
            [title] => Registrant ID
            [description] =>
            [required] => 1
            [type] => text
        )
    [2] => stdClass Object
        (
            [name] => registrantIDType
            [title] => Registrant Type
            [description] =>
            [required] => 1
            [options] => Array
                (
                    [0] => stdClass Object
                        (
                            [value] => ABN
                            [description] => Australian Business Number
                        )
                    [1] => stdClass Object
                        (
                            [value] => ACN
                            [description] => Australian Company Number
                        )
                    [2] => stdClass Object
                        (
                            [value] => OTHER
                            [description] => Other
                        )
                )
            [type] => select
        )
)
```

#### Example of successful eligibility field retrieval for .eco

```
[status] => OK_ELIGIBILITY
[errorMessage] => This extension has eligibility criteria.
[fields] => Array
(
    [0] => stdClass Object
        (
            [name] => eco-agree
            [title] => I confirm I am able to register this domain name as per the eligibility requirements. <br /><sub><i>You can read more on detailed summary of the <a
onclick="window.open('eligibility/summary','window','width=840,height=768,resizable');return false;" id="summary" style="cursor:pointer">.ECO eligibility
criteria</a>.</i></sub>
            [description] => <p>The registration of .ORGANIC requires your confirmation as a registrant to understand and agree with <a
href="http://get.organic/registration_policy" target="_blank">.ORGANIC Registry Policy.</a> By checking the box below, you agree that you fulfil the eligibility criteria as
the registrant.</p>
            [required] => 1
            [policySummary] => <p><b>Eligibility requirements</b></p><p>Anyone can register .organic domain name, however you must ensure the content of the
website is within the guidelines of IFOAM Organics International's Principles of Organic Agriculture, including: :</p><ul><li>Certified organic producers, farmers,
distributors and the like</li><li>Certified organic textile and skincare providers</li><li>Organic restaurants and venues</li><li>Certifiers in the organic
community</li><li>Publications, journalists and bloggers catering to the organic community and industry</li><li>Non-profit, not-for-profit and trade associations that
primarily serve and represent the organic community</li></ul><p>For more details, please go to <a href="http://get.organic/registration_policy" target="_blank">.organic
registration policy</a></p>
            [type] => checkbox
        )
)
```

)  
)

**Example of successful eligibility retrieval with no eligibility**

[status] => OK\_NO\_ELIGIBILITY  
[errorMessage] => The provided extension does not have any eligibility criteria at this time

**Example of failed eligibility retrieval**

[status] => ERR\_NOT\_SUPPORTED  
[errorMessage] => The provided domain name or extension is not supported

# List Domain Categories

This function will allow you to retrieve the list of domain categories under your account.

**Command:** listDomainCategories

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
domainCategories	Array	Array containing list of domain categories
id	String	The ID of a domain category
name	String	The name of a domain category
note	String	The extra note of a domain category
createdDate	dateTime	The date and time that a domain category was created

Example of successful domain categories retrieval (Records exist)
<pre>[status] =&gt; OK [errorMessage] =&gt; Category list successful. [domainCategories] =&gt; Array (   [0] =&gt; Array   (     [id] =&gt;     [name] =&gt; Category Test One     [note] =&gt; Note Test One     [createdDate] =&gt; 2024-01-01 00:00:00   )   [1] =&gt; Array   (     [id] =&gt; 39     [name] =&gt; Category Test Two     [note] =&gt; Note Test Two     [createdDate] =&gt; 2024-02-02 00:00:00   ) )</pre>

Example of successful domain categories retrieval (No records found)
<pre>[status] =&gt; OK [errorMessage] =&gt; Category list successful. [domainCategories] =&gt; Array ( )</pre>

Example of failed domain categories retrieval (Failed authentication)
<pre>[status] =&gt; ERR_LOGIN_FAILED [errorMessage] =&gt; Unable to login to Synergy Wholesale system</pre>

# Create Domain Category

This function will allow you to create a new domain category under your account.

**Command:** createDomainCategory

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
name	Mandatory	Name of the new domain category
note	Optional	Extra note of the new domain category

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
category	Array	Array containing details of the new domain category
id	String	The ID of the new domain category
name	String	The name of the new domain category
note	String	The extra note of the new domain category
createdDate	dateTime	The date and time that the new domain category was created

## Example of successful domain category creation

```
[status] => OK
[errorMessage] => Category created!
[category] => Array
  (
    [id] =>
    [name] => Category Test
    [note] => Some testing notes
    [createdDate] => 2024-01-01 00:00:00
  )
```

## Example of failed domain category creation (Validation errors)

```
[status] => ERR_CREATE_DOMAIN_CATEGORY_FAILED
[errorMessage] => Category Name cannot be empty.
```

## Example of failed domain category creation (Category name duplicated)

```
[status] => ERR_CREATE_DOMAIN_CATEGORY_FAILED
[errorMessage] => A category with the provided name already existed.
```

# Update Domain Category

This function will allow you to update an existing domain category under your account.

**Command:** updateDomainCategory

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
id	Mandatory	Category ID can be obtained from the list domain categories endpoint
name	Mandatory	Updated name of the target domain category
note	Optional	Updated extra note of the target domain category

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
category	Array	Array containing updated details of the target domain category
id	String	The ID of the target domain category
name	String	The updated name of the target domain category
note	String	The updated extra note of the target domain category
createdDate	dateTime	The date and time that the target domain category was created

Example of successful domain category update
<pre>[status] =&gt; OK [errorMessage] =&gt; Category update successful [category] =&gt; Array (   [id] =&gt;   [name] =&gt; Updated Category Test   [note] =&gt; Updated Note   [createdDate] =&gt; 2024-01-01 00:00:00 )</pre>

Example of failed domain category update (Validation errors)
<pre>[status] =&gt; ERR_UPDATE_DOMAIN_CATEGORY_FAILED [errorMessage] =&gt; Category not found.</pre>

Example of failed domain category update (Category name duplicated)
<pre>[status] =&gt; ERR_UPDATE_DOMAIN_CATEGORY_FAILED [errorMessage] =&gt; A category with the provided name already existed.</pre>

# Remove Domain Category

This function will allow you to remove an existing domain category under your account.

**Command:** removeDomainCategory

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
id	Mandatory	Category ID can be obtained from the list domain categories endpoint

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
category	Array	Array containing details of the target domain category
id	String	The ID of the target domain category
name	String	The name of the target domain category
note	String	The extra note of the target domain category
createdDate	dateTime	The date and time that the target domain category was created

## Example of successful domain category removal

```
[status] => OK
[errorMessage] => Category has been removed.
[category] => Array
(
  [id] =>
  [name] => Category Test
  [note] => Testing Note
  [createdDate] => 2024-01-01 00:00:00
)
```

## Example of failed domain category removal (Validation errors)

```
[status] => ERR_REMOVE_DOMAIN_CATEGORY_FAILED
[errorMessage] => Category not found.
```

# Assign Domain Category

This function will allow you to assign a domain to an existing domain category under your account.

**Command:** assignDomainCategory

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
categoryID	Mandatory	Category ID can be obtained from the list domain categories endpoint. This 'categoryID' field will be selected to query the category object as a higher priority. If this field was not provided, 'categoryName' field will be selected as an alternative option
categoryName	Optional	Name of the target domain category. This field will be selected to query the category object as an alternative option if field 'categoryID' was not provided

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful domain category assignment

```
[status] => OK  
[errorMessage] => Domain has been assigned to the category.
```

## Example of failed domain category assignment (Validation errors)

```
[status] => ERR_ASSIGN_DOMAIN_CATEGORY_FAILED  
[errorMessage] => Category not found.
```

# Unassign Domain Category

This function will allow you to unassign a domain from an existing domain category under your account.

**Command:** unassignDomainCategory

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned
categoryId	Mandatory	Category ID can be obtained from the list domain categories endpoint. This 'categoryId' field will be selected to query the category object as a higher priority. If this field was not provided, 'categoryName' field will be selected as an alternative option
categoryName	Optional	Name of the target domain category. This field will be selected to query the category object as an alternative option if field 'categoryId' was not provided

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful domain category unassignment

```
[status] => OK  
[errorMessage] => Domain unassigned from the category.
```

## Example of failed domain category unassignment (Validation errors)

```
[status] => ERR_UNASSIGN_DOMAIN_CATEGORY_FAILED  
[errorMessage] => Category not found.
```

# Initiate Domain Registrant Correction

This function will allow you initiate a domain registrant correction process

**Command:** initiateCorrection

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful initiate domain registrant correction

```
[status] => INITIATE_CORRECTION_SUCCESS  
[errorMessage] => Successfully started new registrant correction process.
```

## Example of failed initiating domain registrant correction (domain is not eligible for correction)

```
[status] => INITIATE_CORRECTION_FAILED  
[errorMessage] => Domain is not eligible for a registrant correction.
```

# Check Domain Registrant Correction

This function will allow you to check for a pending domain registrant correction

**Command:** checkCorrection

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
isEligibleForCorrection	Boolean	A boolean (true/false) indicating whether the domain is eligible for a correction or not. For example, a domain cannot have a correction if there are any already in progress
isPendingCorrection	Boolean	A boolean indicating whether there is a pending correction or not
pendingCorrectionStatus	String	The pending correction status (possible values: awaiting_new_legal_entity, awaiting_existing_agreement, pending_staff_approval, auto_approve_failed, completed)
pendingCorrectionLink	String	The link to the pending correction (if any)

## Example of domain registrant correction check with in progress correction

```
[status] => OK
[errorMessage] => Successfully retrieved correction status for domain.
[isEligibleForCorrection] => false
[isPendingCorrection] => true
[pendingCorrectionStatus] => awaiting_new_legal_entity
[pendingCorrectionLink] => http://interface.domain.com/registrant-correction/j117...
```

## Example of domain registrant correction check with no corrections

```
[status] => OK
[errorMessage] => Successfully retrieved correction status for domain.
[isEligibleForCorrection] => true
[isPendingCorrection] => false
[pendingCorrectionStatus] => null
[pendingCorrectionLink] => null
```

# Resend Domain Registrant Correction Email

This function will resend domain registrant correction email

**Command:** resendCorrectionEmail

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
domainName	Mandatory	Domain name to be actioned

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful resending domain registrant correction email

```
[status] => RESEND_EMAIL_SUCCESS  
[errorMessage] => Successfully resent Registrant Correction email to the registrant.
```

## Example of failed resending domain registrant correction email (domain is not pending a correction)

```
[status] => ERR_NO_REGISTRANT_CORRECTION  
[errorMessage] => This domain is not pending a registrant correction.
```

# Hosting Services

The following commands allow for acquisition and management of web & email hosting services

# List Hosting Services

The following command returns a paginated list of hosting services (and information) within the Synergy Wholesale partner account. **Note:** There is a maximum of 1000 services returned per page.

**Command:** listHosting

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
page	Optional	An integer specifying the page you would like to return
limit	Optional	An integer specifying the result limit (500 are returned if no value supplied)
status	Optional	Allows you to filter services by status ('Inactive', 'Active', 'Terminated', 'Suspended', 'Pending Cancellation', 'Cancelled', 'Pending Completion', 'Suspended By Staff', 'Pending Upgrade'). <b>Note:</b> Only one value can be supplied at a time.

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the failed command
domainList	array	Contains the nested hosting info for each hosting service
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the failed command
domain	string	Domain name of the hosting service
hoid	string	The unique hosting service identifier
username	string	The username of the hosting service
password	string	Password for the hosting service
server	string	Name of the server the hosting service is on
dedicatedIPv4	string	Dedicated IP Address of the service (if any otherwise null)
serviceStatus	string	Status of the hosting service
planID	string	The plan assigned to the service (please refer to the "Hosting Plans Information" table)
plan	string	The name of the plan the hosting service is on
billingPeriod	string	Period on which the hosting service is billed
nextRenewalDue	string	Date the service is due to be billed
locationID	string	The geographical location ID of the hosting service (please refer to the "Hosting Locations Information" table)
city	string	Name of the city in which the server is hosted in
country	string	Name of the country in which the server is hosted in
serverIPAddress	string	Public IP Address of the server the hosting service is on
page	integer	The current page of the returned results
limit	integer	The limit on the results set that have been returned

#### Example of successful list hosting services command

```
[status] => OK
[page] => 1
[limit] => 5
[hoidList] => Array
(
  [0] => stdClass Object
    (
      [status] => OK
      [hoid] => 59d1c0f88976f
      [domain] => thesuperpotatoclub.net.au
      [username] => thesuperpo
      [password] => v[0d^gnDs3]5}
      [serviceStatus] => Active
      [dedicatedIPv4] => 155.55.55.55
      [server] => WSH1-1E-SYD
      [plan] => EW1
      [billingPeriod] => monthly
      [nextRenewalDue] => 2017-10-31 23:59:59
      [city] => Sydney
      [country] => Australia
      [dedicatedIPv4] => 155.55.55.56
      [country] => Australia
    )
)
```

#### Example of failed list hosting services command where a hosting service does not exist

```
[status] => OK
[page] => 1
[limit] => 5
[domainList] => Array
(
  [0] => stdClass Object
    (
      [domainName] => thesuperpotatoclub.net.au
      [status] => ERR_HOSTING_SERVICE_NOT_FOUND
      [errorMessage] => Hosting service does not exist
    )
)
```

#### Example of failed list hosting services command

```
[status] => ERR_OCCURED
[errorMessage] => Failed to list hosting services. Please try again later
```

# Hosting Service Query

The following command allows you to obtain information for a hosting service.

**Command:** hostingGetService

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	The 'identifier' field has replaced the 'hoid' field and will accept either the 'hoid' of a hosting account or the email address of an email hosting service. If this is empty, it will default to use the 'hoid' variable value to prevent any breaking changes.
hoid (Deprecated)	Optional	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
domain	String	Domain name of requested hosting service
username	String	Username of the requested hosting service
email	String	Email of the requested hosting service
server	String	Server used for the requested hosting service
product	String	Name of the product
plan	String	Plan name of the requested hosting service
billingPeriod	String	Billing period for the service, when the service renews
nextRenewalDue	String	Next renewal date of the service
password	String	Password for the service
dedicatedIPv4	String	If the service has a dedicated IP it will be returned
dkim	String	The DKIM (Domain Keys Identified Mail) for an email hosting service
mxrecords	String	A JSON encoded array containing the mx record and it's priority as the key and value respectively. E.g. mx.test.com => 10.
activeSync	String	States whether Active Sync is "enabled" or "disabled"
mailboxUsage	String	The usage of the Email hosting service in GB

## Example of successful execution of the function

```
[status] => Active
[errorMessage] => Hosting Get Service Completed Successfully
[domain] => apimootest.com
[username] => apimootest
[server] => ws2-2e-mel.hostyourservices.net
[plan] => EW2
[billingPeriod] => monthly
[nextRenewalDue] => 2014-06-01
```

## Example of a failed execution of the function

```
[status] => ERR_DOMAININFO_FAILED
[errorMessage] => Domain Info Failed - Domain Does Not Exist
[domainName] => nonexistentdomain.com.au
```

[domain\_status] => Domain does not exist

# Bulk Hosting Service Query

The following command allows you to obtain information for multiple hosting services where the HO-IDs are provided.

**Command:** bulkHostingInfo

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
hoidList	Mandatory	An array of HO-IDs for a list of services
limit	Optional	An integer specifying the result limit (1000 are returned if no value supplied)

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
domainList	Array	Contains the nested hosting info for each HO-ID supplied
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
domain	String	Domain name of the hosting service
hoid	String	The unique hosting service identifier
username	String	The username of the hosting service
password	String	Password for the hosting service
server	String	Name of the server the hosting service is on
dedicatedIPv4	String	Dedicated IP Address of the service (if any otherwise null)
serviceStatus	String	Status of the hosting service
planID	String	The plan assigned to the service (please refer to the "Hosting Plans Information" table)
plan	String	The name of the plan the hosting service is on
billingPeriod	String	Period on which the hosting service is billed
nextRenewalDue	String	Date the service is due to be billed
locationID	String	The geographical location ID of the hosting service (please refer to the "Hosting Locations Information" table)
city	String	Name of the city in which the server is hosted in
country	String	Name of the country in which the server is hosted in
serverIPAddress	String	Public IP Address of the server the hosting service is on

#### Example of successful bulk hosting service query command

```
[status] => OK
[hoidList] => Array
(
  [0] => stdClass Object
  (
    [status] => OK
    [hoid] => 59d1c0f88976f
    [domainName] => thesuperpotatoclub.net.au
    [username] => thesuperpo
    [password] => v[0d^gnDs3]5}
    [serviceStatus] => Active
    [dedicatedIPv4] => 155.55.55.55
    [server] => WSH1-1E-SYD
    [plan] => EW1
    [billingPeriod] => monthly
    [nextRenewalDue] => 2017-10-31 23:59:59
    [city] => Sydney
    [country] => Australia
    [dedicatedIPv4] => 155.55.55.56
    [country] => Australia
  )
)
```

#### Example of failed bulk hosting service query command where a hosting service does not exist

```
[status] => OK
[domainList] => Array
(
  [0] => stdClass Object
  (
    [domainName] => thesuperpotatoclub.net.au
    [status] => ERR_HOSTING_SERVICE_NOT_FOUND
    [errorMessage] => Hosting service does not exist
  )
)
```

#### Example of failed bulk hosting service query command

```
[status] => ERR_OCCURED
[errorMessage] => Failed to perform bulk hosting info. Please try again later
```

# Purchase Hosting Service

The following command allows you to purchase a new hosting service for both Web hosting and Email hosting.

**Command:** hostingPurchaseService

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation.
apiKey	Mandatory	API Key can be obtained from the wholesale system interface.
planName	Mandatory	The name of the hosting/email plan you created inside the Synergy Wholesale System.
domain	Mandatory	Domain name for the new hosting service.
email	Mandatory	Email address for the new hosting service.
username	Optional	The username for your hosting service.
password	Optional	The password for your hosting service, For hosting, please note this is still subject to cPanel password strength requirements.
firstName	Optional	The first name for the new email hosting service.
lastName	Optional	The last name for the new email hosting service.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
username	String	Username of the newly created service
domain	String	Domain name of the hosting service
email	String	Email of the hosting service
password	String	Password assigned to the hosting service

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Hosting Purchase Service Completed Successfully
[username] => moocown0
[domain] => moocownugs.com
[password] => c)2h2Q!}W352H2
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGSUSPENDSERVICE_FAILED
[errorMessage] => Hosting Suspend Service Failed - Unable to suspend account - Please contact technical support for assistance with this hosting service
```

# Suspend Hosting Service

The following command allows you to suspend the requested hosting service.

**Command:** hostingSuspendService

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
reason	Mandatory	Reason for suspending the service
identifier	Mandatory	The 'identifier' field has replaced the 'hoid' field and will accept either the 'hoid' of a hosting account or the email address of an email hosting service. If this is empty, it will default to use the 'hoid' variable value to prevent any breaking changes.
hoid ( <b>Deprecated</b> )	Optional	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Suspend Service Completed Successfully
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGSUSPENDSERVICE_FAILED  
[errorMessage] => Hosting Suspend Service Failed - Unable to suspend account - Please contact technical support for assistance with this hosting service
```

# Unsuspend Hosting Service

The following command allows you to unsuspend the requested hosting service.

**Command:** hostingUnsuspendService

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	The 'identifier' field has replaced the 'hoid' field and will accept either the 'hoid' of a hosting account or the email address of an email hosting service. If this is empty, it will default to use the 'hoid' variable value to prevent any breaking changes.
hoid (Deprecated)	Optional	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Unsuspend Service Completed Successfully
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGUNSUSPENDSERVICE_FAILED  
[errorMessage] => Hosting Unsuspend Service Failed - Unable to Unsuspend account - Please contact technical support for assistance with this hosting service
```

# List Hosting Packages

The following command allows you to retrieve a list of your configured packages.

**Command:** hostingListPackages

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
product	Optional	The type of the product you wish to retrieve packages for. Valid values: Custom Hosting and Email Hosting. If not specified will return packages for all product types.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
packages	Array	Contains a list of all configured packages.
name	String	The name of the package.
product	String	The product type this package is for. Valid values: Custom Hosting or Email Hosting
price	Decimal	The recurring price of the package.

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Successfully retrieved package list
[packages] => Array
(
  [0] => stdClass Object
  (
    [name] => Basic Plan
    [product] => Custom Hosting
    [price] => 6.50
  )
  [1] => stdClass Object
  (
    [name] => Email Small
    [product] => Email Hosting
    [price] => 0.75
  )
)
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGLISTPACKAGES_FAILED
[errorMessage] => Hosting List Packages Failed - Unable to retrieve package list for account
```

# Change Hosting Password

The following command allows you to change the password of the requested hosting service.

**Command:** hostingChangePassword

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
newPassword	Mandatory	New password for the hosting service. If set to AUTO we will generate a new password
identifier	Mandatory	The 'identifier' field has replaced the 'hoid' field and will accept either the 'hoid' of a hosting account or the email address of an email hosting service. If this is empty, it will default to use the 'hoid' variable value to prevent any breaking changes.
hoid ( <b>Deprecated</b> )	Optional	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
password	String	New password set on the hosting service

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Change Password Completed Successfully  
[password] => {dRWf9]uGg9^u(
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGCHANGEPASSWORD_FAILED  
[errorMessage] => Hosting Change Password Failed - Unable To Update Password On The Hosting Server
```

# Change Hosting Package

The following command allows you to change the plan of the requested hosting service.

**Command:** hostingChangePackage

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
newPlanName	Mandatory	New plan name for the service. Referenced in the table "Hosting Plan Information"
identifier	Mandatory	The 'identifier' field has replaced the 'hoid' field and will accept either the 'hoid' of a hosting account or the email address of an email hosting service. If this is empty, it will default to use the 'hoid' variable value to prevent any breaking changes.
hoid ( <b>Deprecated</b> )	Optional	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
newAmount	String	This is the amount that will be charged per month for the new plan

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Change Package Completed Successfully  
[newAmount] => 1.95
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGCHANGEPACKAGE_FAILED  
[errorMessage] => Hosting Change Package Failed - Cannot upgrade to the same plan as current one
```

# Enable Temporary URL

The following command allows you to enable the temporary viewing URL to see a website that has not been delegated to the correct name servers.

**Command:** hostingEnableTempUrl

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
hoid	Mandatory	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
tempURL	String	This is the temporary URL that can be used for viewing the website

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Temporary URL enabled successfully  
[tempUrl] => http://112.140.176.57/~justtest5
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGENABLETEMPURL_FAILED  
[errorMessage] => Unable to enable temporary URL - <reason>
```

# Disable Temporary URL

The following command allows you to disable the temporary viewing URL used to see a website that has not been delegated to the correct name servers.

**Command:** hostingDisableTempUrl

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
hoid	Mandatory	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

Example of successful execution of the function
[status] => OK [errorMessage] => Temporary URL disabled successfully

Example of a failed execution of the function
[status] => ERR_HOSTINGDISABLETEMPURL_FAILED [errorMessage] => Unable to disable temporary URL - <reason>

# Check IP Address for Firewall Block

The following command allows you to check an IP address against the server firewall to see if it access is blocked

**Command:** hostingCheckFirewall

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
hoid	Mandatory	Hosting Service ID
ipAddress	Mandatory	The IP address to check if blocked

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
Blocked	String (Boolean)	Boolean (true/false) to advise if supplied IP address is blocked or unblocked

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Check Firewall Successful!  
[blocked] => false
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGCHECKFIREWALL_FAILED  
[errorMessage] => Hosting Check Firewall Failed - <reason>
```

# Unblock IP Address from Firewall Block

The following command allows you to request an IP address to be removed from the firewall block

**Command:** hostingUnblockFirewall

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
hoid	Mandatory	Hosting Service ID
ipAddress	Mandatory	The IP address to remove from the firewall block

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
Blocked	String (Boolean)	Boolean (true/false) to advise if supplied IP address was actually blocked or unblocked

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Unblock Firewall Successful!  
[blocked] => false
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGUNBLOCKFIREWALL_FAILED  
[errorMessage] => Hosting Unblock Firewall Failed - <reason>
```

# Recreate Hosting Service

The following command allows you to recreate the service like it was a new account.

**Command:** hostingRecreateService

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
hoid	Mandatory	Hosting Service ID
newPassword	Mandatory	New password for the hosting service. If set to AUTO we will generate a new password

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Service Successfully Recreated
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGRECREATE_FAILED  
[errorMessage] => Hosting Recreate Failed – Domain name exists already
```

# Terminate Hosting Service

The following command allows you to terminate the hosting service which will end all billing and delete all data.

**Command:** hostingTerminateService

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	The 'identifier' field has replaced the 'hoid' field and will accept either the 'hoid' of a hosting account or the email address of an email hosting service. If this is empty, it will default to use the 'hoid' variable value to prevent any breaking changes.
hoid (Deprecated)	Optional	Hosting Service ID

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Hosting Service Successfully Terminated
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGRECREATE_FAILED  
[errorMessage] => Hosting Terminate Failed – Domain name exists already
```

# Generate Hosting Login Link

The following command allows you to generate a login link to the hosting service.

**Command:** hostingGetLogin

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	The 'identifier' field will accept either the 'hoid' of a hosting account or the email address of an email hosting service.

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the failed command
url	String	URL to log directly into the service

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Hosting Get Login Completed Successfully
[url] => https://server.com:2083/cpsess1308433669/login/?session=123%123%3acreate_user_session%1233
```

## Example of a failed execution of the function

```
[status] => ERR_HOSTINGGETLOGIN_FAILED
[errorMessage] => Hosting Get Login Failed - Unable to get account
```

# **SSL Certificates**

The following commands allow for acquisition and management of SSL certificates

# Retrieve SSL Product Information & Pricing

The following command allows you to return the SSL Product ID, Product Name, Product Description and Price.

**Command:** getSSLPricing

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
errorMessage	string	Human readable description on status of command
pricing	array	SSL product ID, product Name, product description and price returned in an array

## Example of successful retrieval of SSL product information and pricing

```
(
  [status] => OK
  [errorMessage] => Pricing has been retrieved
  [pricing] => Array
  (
    [0] => stdClass Object
    (
      [productID] => 55
      [productName] => Trustwave Domain Validated SSL - 1 Year
      [productDescription] => Trustwave Domain Validated SSL - 1 Year
      [remoteProductType] => DV
      [price] => XX.XX
    )
  )
)
```

## Example of failed retrieval of SSL product information and pricing

```
(
  [status] => ERR_RESELLER_NOT_AUTHORISED
  [errorMessage] => Reseller is not authorized to perform this function
)
```

## Query Certificate Status

The following command allows you to obtain information for a SSL certificate you've recently purchased.

**Command:** SSL\_getSSLCertificate

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
certStatus	string	SSL certificate status with Trustwave
cer	string	SSL certificate
p7b	string	SSL certificate provided in Windows format (IIS)
certID	string	Certificate ID provided to ensure you've requested the correct SSL
commonName	string	Domain name related to the SSL certificate
expireDate	string	Expiry date of the SSL if you wish to send renewal notices

# Query Certificate Status (simple)

Command: SSL\_getCertSimpleStatus

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
certStatus	string	Status of the certificate in the wholesale system
commonName	string	Domain name of the SSL certificate

### Example of successful SSL certificate query

```
(  
  [status] => OK  
  [certStatus] => ACTIVE  
  [commonName] => exampledomain.com.au  
)
```

### Example of failed SSL certificate query

```
(  
  [status] => ERR_RESELLER_NOT_AUTHORISED  
  [errorMessage] => Reseller is not authorized to perform any actions on this SSL Certificate  
)
```

# Generate Certificate Signing Request (CSR)

The following command allows you generate the certificate signing request (CSR) to be used when purchasing an SSL certificate using the purchase command.

**Command:** SSL\_generateCSR

You must pass to the API the following variables to generate certificate signing request (CSR).

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
numOfYears	Mandatory	Number of years the CSR is being generated
city	Mandatory	City the certificate will be issued
state	Mandatory	State the certificate will be issued
country	Mandatory	Country the certificate will be issued
organisation	Mandatory	Organisation name the certificate will be issued
organisationUnit	Mandatory	Organisation unit (department) the certificate will be issued
commonName	Mandatory	Domain name of the SSL certificate to be used
emailAddress	Mandatory	Email address assigned to the certificate signing request

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
privKey	string	Private key to be used in combination with the issued certificated
selfSignCrt	string	Self signed certificate (if required)
csr	string	Certificate Signing Request (CSR)

## Example of successful CSR generation

```
(
  [status] => OK
  [privKey] => -----BEGIN RSA PRIVATE KEY-----
               -----END RSA PRIVATE KEY-----
  [selfSignCrt] => -----BEGIN CERTIFICATE-----
                  -----END CERTIFICATE-----
  [csr] => -----BEGIN CERTIFICATE REQUEST-----
          -----END CERTIFICATE REQUEST-----
)
```

## Example of failed CSR generation

```
(
  [status] => ERR_VAR_EMPTY
  [errorMessage] => One of the required variables are missing or empty
)
```

# Decode Certificate Signing Request (CSR)

The following command allows you decode the certificate signing request (CSR) you've provided to ensure all the details are correct before purchasing the SSL.

**Command:** SSL\_decodeCSR

You must pass to the API the following variables to decode the provided certificate signing request (CSR).

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
csr	Mandatory	Certificate signing request

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
commonName	string	Domain name for the certificate
organisationUnit	string	Organisation department the SSL will be issued
organisation	string	Organisation name that the SSL will be issued
state	string	State the SSL will be issued
city	string	City/Suburb the SSL will be issued
country	string	Country the SSL will be issued
subjectAltNames	array	The SAN's for the CSR

## Example of successful CSR decode

```
(
  [status] => OK
  [state] => VIC
  [country] => AU
  [city] => Melbourne
  [organisationUnit] => Technical Department
  [organisation] => John Doe Pty Ltd
  [commonName] => www.exampledomain.com
  [subjectAltNames] =>
)
```

## Example of failed CSR decode

```
(
  [status] => ERR_DECODECSR_FAILED
  [errorMessage] => Failed to parse CSR
)
```

# Purchase New SSL Certificate

The following command allows you to purchase a new SSL certificate via the API.

**Command:** `SSL_purchaseSSLCertificate`

You must pass to the API the following variables to purchase an SSL certificate.

When requesting an SSL we will automatically send emails to the following addresses for approval of the SSL certificate

[admin@domainname.com](mailto:admin@domainname.com)  
administrator@domainname.com  
webmaster@domainname.com  
hostmaster@domainname.com

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
csr	Mandatory	Certificate request that you required signed
productID	Mandatory	Please refer to the table below "SSL Product IDs"
firstName	Mandatory	First name of the customer purchasing the SSL
lastName	Mandatory	Last name of the customer purchasing the SSL
emailAddress	Mandatory	Email address of the customer purchasing the SSL
address	Mandatory	Physical of the customer purchasing the SSL
city	Mandatory	City/suburb of the customer purchasing the SSL
state	Mandatory	State of the customer purchasing the SSL
postCode	Mandatory	Post code of the customer purchasing the SSL
country	Mandatory	Country code of the customer purchasing the SSL (eg. AU)
phone	Mandatory	Phone number of the customer purchasing the SSL Note: Please use international format. Eg +61.390245383
fax	Mandatory	Fax number of the customer purchasing the SSL

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
certStatus	string	SSL certificate status with Trustwave
certID	string	SSL certificate ID used for reference to check the status
commonName	string	Domain name used for the SSL certificate

Example of successful SSL purchase
<pre>(   [status] =&gt; OK   [certStatus] =&gt; PENDING   [certID] =&gt; 57291   [commonName] =&gt; wholesalesystem.com.au )</pre>

Example of failed SSL certificate purchase
<pre>(   [status] =&gt; ERR_PURCHASE_CERT_FAILED   [errorMessage] =&gt; buyCertificate: Failed to send the request )</pre>

# Reissue SSL Certificate

The following command allows you re-issue the SSL certificate using a new certificate signing request (CSR)

**Command:** SSL\_reissueCertificate

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response
newCSR	Mandatory	New certificate signing request (CSR) to be used for reissue.

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
cer	string	SSL certificate
p7b	string	SSL certificate provided in Windows format (IIS)
newCertID	string	New wholesale system certificate ID

```
Example of successful SSL reissue  
  
(  
  [status] => OK  
  [newCertID] => XXX  
  
  [cer] => -----BEGIN CERTIFICATE-----  
           SSL certificate will be provided here  
           -----END CERTIFICATE-----  
  
  [p7b] => -----BEGIN PKCS7-----  
           SSL certificate will be provided here  
           -----END PKCS7-----  
)
```

```
Example of failed SSL reissue  
  
(  
  [status] => ERR_REISSUE_FAILED  
  [errorMessage] => Certificate not found  
)
```

```
Example 2 of failed SSL reissue  
  
(  
  [status] => ERR_REISSUE_FAILED  
  [errorMessage] => Failed to parse CSR  
)
```

# Cancel SSL Certificate

The following command allows you cancel an SSL certificate that has not been processed (eg. still pending approval).

**Command:** SSL\_cancelSSLCertificate

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
certStatus	string	Status from the cancelled certificate

**Example of successful cancellation**

```
(  
  [status] => OK  
  [certStatus] => REFUNDED  
)
```

**Example of failed cancellation**

```
(  
  [status] => ERR_CANCELCERTIFICATE_FAILED  
  [errorMessage] => SSL Certificate is in an unsupported state  
)
```

# Renew SSL Certificate

The following command allows you renew an SSL certificate.

**Command:** SSL\_renewSSLCertificate

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response
firstName	Mandatory	First name of the customer purchasing the SSL
lastName	Mandatory	Last name of the customer purchasing the SSL
emailAddress	Mandatory	Email address of the customer purchasing the SSL
address	Mandatory	Physical of the customer purchasing the SSL
city	Mandatory	City/suburb of the customer purchasing the SSL
state	Mandatory	State of the customer purchasing the SSL
postCode	Mandatory	Post code of the customer purchasing the SSL
country	Mandatory	Country code of the customer purchasing the SSL (eg. AU)
phone	Mandatory	Phone number of the customer purchasing the SSL Note: Please use international format. Eg +61.390245383
fax	Mandatory	Fax number of the customer purchasing the SSL

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
certStatus	string	Status of the renewal request submitted
certID	string	New certificate ID upon successful renewal
commonName	string	Domain name of the SSL certificate

Example of successful renewal
<pre>(   [status] =&gt; OK   [certStatus] =&gt; PENDING   [certID] =&gt; XXX   [commonName] =&gt; exampledomain.com.au )</pre>

Example of failed renewal
<pre>(   [status] =&gt; ERR_CERT_NOT_WITHIN_EXPIRY   [errorMessage] =&gt; SSL Certificate Is Not Within The Required 90 Days Expiry )</pre>

# Resend Approval Email

The following command allows you resend the approval email for an SSL certificate.

**Command:** SSL\_resendDVEmail

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response

Example of successful approval email resent
<pre>(   [status] =&gt; OK )</pre>

Example of failed approval email resent
<pre>(   [status] =&gt; ERR_INVALID_CERT_STATE   [errorMessage] =&gt; Certificate id supplied is in an invalid state )</pre>

# Resend Completion Certificate Email

The following command allows you resend the original completed certificate email to the customer. This is helpful should your customer lose the details of their SSL and you need to provide the information again.

**Command:** SSL\_resendIssuedCertificateEmail

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed

Example of successful completed email resent
<pre>(   [status] =&gt; OK )</pre>

Example of failed completed email resent
<pre>(   [status] =&gt; ERR_INVALID_CERT_STATE   [errorMessage] =&gt; Certificate id supplied is in an invalid state )</pre>

# Force Check TXT Record for DNS Domain Validation

The following command allows you to force a check, and subsequently retrieve, the TXT record when utilising domain validation through the DNS Validation method for SSL Certificates (where supported by certificate type).

**Command:** SSL\_checkTxtCodes

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message on outcome
validationStatus	string	The validation status for the certificate as a whole as returned by the SSL vendor
domains	array	The domain names (SANs) in the list for the specific SSL certificate (depending on what was provided in the initial CSR, where multi-SAN certificates are supported by the SSL vendor. See example for specific strings returned as part of this array.

## Example of successful TXT Record domain name validation

```
(
  [status] => OK
  [errorMessage] => SSL Check successful
  [validationStatus] => PENDING
  [domains] => [
    [0] => [
      [status] => VALID
      [domain] => synergyisawesome.com.au
    ]
    [1] => [
      [status] => PENDING
      [domain] => www.synergyisawesome.com.au
    ]
  ]
)
```

## Example of failed TXT Record domain name validation

```
(
  [status] => ERR_RESELLER_NOT_AUTHORISED
  [errorMessage] => Reseller is not authorized to perform any actions on this SSL Certificate
)
```

## Example 2 of failed (all SANs 'pending') TXT Record domain name validation

```
(
  [status] => OK
  [errorMessage] => SSL Check successful
  [validationStatus] => PENDING
  [domains] => [
    [0] => [
      [status] => PENDING
      [domain] => mysynergy-test.com.au
    ]
    [1] => [
      [status] => PENDING
      [domain] => www.mysynergy-test.com.au
    ]
  ]
)
```

# List All Certificates

The following command allows you obtain a list of all SSL certificates related to your account.

**Command:** SSL\_listAllCerts

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
certID	string	ID used for obtaining the details at a later time
productID	string	Product ID to determine the type of SSL certificate
csr	string	Certificate Signing Request (CSR)
cer	string	SSL Certificate
p7b	string	SSL certificate provided in Windows format (IIS)
commonName	string	Domain name of the SSL certificate
status	string	Certificate status
startDate	string	Issued Date
expireDate	string	Expiry Date
orderDate	string	Date the certificate was ordered
newCertID	string	New certificate ID in cases where the SSL was reissued

```
Example of successful list certificates

(
    [certID] => XXX
    [productID] => XXX

    [csr] => -----BEGIN CERTIFICATE REQUEST-----
            SSL signing request will be provided here
            -----END CERTIFICATE REQUEST-----

    [cer] => -----BEGIN CERTIFICATE-----
            SSL certificate will be provided here
            -----END CERTIFICATE-----

    [p7b] => -----BEGIN PKCS7-----
            SSL certificate will be provided here
            -----END PKCS7-----

    [commonName] => exampledomainname.com
    [status] => ACTIVE
    [startDate] => 2011-01-01T23:59:48-05:00
    [expireDate] => 2012-01-01T23:59:48-05:00
    [orderDate] => 2011-01-01T23:59:48-05:00
    [newCertID] => 584992
)
```

# Get Domain Beacon

The following command allows you retrieve the domain beacon that is used for SSL validation.

The domain beacon is used for verification of premium SSL certificates to prove ownership of the domain and ensure the requester has access to the domain in question. The domain beacon file must be saved as the filename returned from the API request and the 'beacon' saved in the file.

**Command:** SSL\_getDomainBeacon

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response
domainName	Mandatory	Domain name of the SSL certificate

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed
filename	string	File to be saved as this name for verification
beacon	string	File needs to contain this text for the verification process
domain	string	Domain name of the successful beacon request

```
Example of successful domain beacon  
  
(  
  [status] => OK  
  [filename] => trustwave_8938980b1b394e4aac79acdee40a184d.html  
  [beacon] => 8938980b-1b39-4e4a-ac79-acdee40a184d  
  [domain] => exampledomain.com.au  
)
```

```
Example of failed domain beacon  
  
(  
  [status] => ERR_GETDOMAINBEACON_FAILED  
  [errorMessage] => Unable to retrieve the domain beacon code for the supplied certificate id  
)
```

# Check Domain Beacon

Upon requesting the domain beacon from 'SSL\_getDomainBeacon' this function will then process the SSL for validation against the certificate ID supplied.

**Note:** Ensure you have completed the beacon creation file from 'SSL\_getDomainBeacon' by creating the filename in the root directory of your web account and that it is visible by browsing to the filename before continuing with this check process

**Command:** SSL\_checkDomainBeacon

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
certID	Mandatory	Certificate ID provided in the API SSL purchase response
domainName	Mandatory	Domain name of the SSL certificate

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of success/failed

Example of successful domain beacon check
<pre>(   [status] =&gt; OK )</pre>

Example of failed domain beacon check
<pre>(   [status] =&gt; ERR_GETDOMAINBEACON_FAILED   [errorMessage] =&gt; Unable to retrieve the domain beacon code for the supplied certificate id )</pre>

# **Services & Product Information**

The following contains additional information relating and pertaining to the services and products as referenced in sections of our API Documentation

## SSL Certificate Product IDs

When purchasing an SSL via our API you need to use the following product IDs as mentioned during the SSL purchase function. Please make note of the years for each product to ensure you only order the correct SSL certificate.

Product ID	SSL Certificate
55	Trustwave Domain Validated SSL - 1 Year
56	Trustwave Domain Validated SSL - 2 Year
57	Trustwave Domain Validated SSL - 3 Year
58	Trustwave Premium SSL - 1 year
59	Trustwave Premium SSL - 2 year
60	Trustwave Premium SSL - 3 year
61	Trustwave Enterprise SSL - 1 Year
62	Trustwave Enterprise SSL - 2 Year
63	Trustwave Enterprise SSL - 3 Year
64	Trustwave Premium Wildcard SSL - 1 Year
65	Trustwave Premium Wildcard SSL - 2 Year
66	Trustwave Premium Wildcard SSL - 3 Year
67	Trustwave Premium EV SSL - 1 Year
68	Trustwave Premium EV SSL - 2 Year

# Hosting Plans Information

The following command allows you to obtain the account balance.

Plan	Specifications
EW1	<p data-bbox="459 412 1324 436">Please refer to the Pricing information page within the Wholesale System for individual plan information.</p> <p data-bbox="630 454 1153 479"><i>[Login to Synergy Wholesale System] -&gt; [Financial] -&gt; [Pricing]</i></p>
EW2	
EW3	
EW4	
EW5	
BW1	
BW2	
BW3	
BW4	
BW5	

## Hosting Locations Information

The purchase process of a new hosting service requires the location ID to be provided. You're able to choose from the below locations for the account to be provisioned using the 'Location ID'.

Location	Specifications
SYDNEY	NEXTDC S1 – Sydney, Australia
MELBOURNE	NEXTDC M1 – Melbourne, Australia

# DNS Configuration Information

Provided below is information on the DNS included with all domain names.

## DNS Config Values

The following values are required for enabling forwarding, parking or DNS hosting a domain via the Update Name Servers function value option of 'dnsConfig'. The value of '1' is not required when updating a domain to custom name servers as its assumed you don't wish to use forwarding, parking or DNS hosting.

Only domain names which are within your wholesale account are able to be configured for DNS or forwarding, third party domain names cannot be added.

**Note:** When swapping between Email/Web Forwarding and DNS Hosting any existing records are deleted immediately.

Value	Description
1	Custom Name Servers
2	Email/Web Forwarding
3	Parked
4	DNS Hosting

## Name Servers

The following name servers are used when using forwarding, parking or DNS hosting. You cannot assign the name servers manually to a domain as the DNS zone isn't created during the process, you must use the Update Name Servers function.

ns1.nameserver.net.au

ns2.nameserver.net.au

ns3.nameserver.net.au

## URL Forwarding Types

The configuration for adding a new URL forward is complex based on the options you choose to use. Below is an explanation of the 'type' variable for creating a new forwarder.

Value	Description
C	Cloaking
H	HTML Redirect
P	301 Redirect - Permanent
T	302 Redirect - Temporary

# Country List

The registries required the country code to be used for registrations, transfers and contact updates.

Country Code	Country	Country Code	Country
AC	Ascension	LA	Laos
AD	Andorra	LB	Lebanon
AE	United Arab Emirates	LC	Saint Lucia
AF	Afghanistan	LI	Liechtenstein
AG	Antigua and Barbuda	LK	Sri Lanka
AI	Anguilla	LR	Liberia
AL	Albania	LS	Lesotho
AM	Armenia	LT	Lithuania
AN	Netherlands Antilles	LU	Luxembourg
AO	Angola	LV	Latvia
AQ	Peter I Island	LY	Libya
AQ	Ross Dependency	MA	Morocco
AQ	Australian Antarctic Territory	MC	Monaco
AQ	British Antarctic Territory	MD	Moldova
AQ	Queen Maud Land	MD	Pridnestrovie (Transnistria)
AR	Argentina	ME	Montenegro
AS	American Samoa	MG	Madagascar
AT	Austria	MH	Marshall Islands
AU	Australia	MK	Macedonia
AW	Aruba	ML	Mali
AX	Aland	MM	Myanmar (Burma)
AZ	Nagorno-Karabakh	MN	Mongolia
AZ	Azerbaijan	MO	Macau
BA	Bosnia and Herzegovina	MP	Northern Mariana Islands
BB	Barbados	MQ	Martinique
BD	Bangladesh	MR	Mauritania
BE	Belgium	MS	Montserrat
BF	Burkina Faso	MT	Malta
BG	Bulgaria	MU	Mauritius
BH	Bahrain	MV	Maldives
BI	Burundi	MW	Malawi
BJ	Benin	MX	Mexico
BM	Bermuda	MY	Malaysia
BN	Brunei	MZ	Mozambique
BO	Bolivia	NA	Namibia
BR	Brazil	NC	New Caledonia
BS	Bahamas, The	NE	Niger
BT	Bhutan	NF	Norfolk Island
BV	Bouvet Island	NG	Nigeria
BW	Botswana	NI	Nicaragua

BY	Belarus	NL	Netherlands
BZ	Belize	NO	Norway
CA	Canada	NP	Nepal
CC	Cocos (Keeling) Islands	NR	Nauru
CD	Congo, (Congo Kinshasa)	NU	Niue
CF	Central African Republic	NZ	New Zealand
CG	Congo, (Congo Brazzaville)	OM	Oman
CH	Switzerland	PA	Panama
CI	Cote d'Ivoire (Ivory Coast)	PE	Peru
CK	Cook Islands	PF	French Polynesia
CL	Chile	PF	Clipperton Island
CM	Cameroon	PG	Papua New Guinea
CN	China, People's Republic of	PH	Philippines
CO	Colombia	PK	Pakistan
CR	Costa Rica	PL	Poland
CU	Cuba	PM	Saint Pierre and Miquelon
CV	Cape Verde	PN	Pitcairn Islands
CX	Christmas Island	PR	Puerto Rico
CY	Cyprus	PT	Portugal
CY	Northern Cyprus	PW	Palau
CZ	Czech Republic	PY	Paraguay
DE	Germany	QA	Qatar
DJ	Djibouti	RE	Reunion
DK	Denmark	RO	Romania
DM	Dominica	RS	Serbia
DO	Dominican Republic	RU	Russia
DZ	Algeria	RW	Rwanda
EC	Ecuador	SA	Saudi Arabia
EE	Estonia	SB	Solomon Islands
EG	Egypt	SC	Seychelles
ER	Eritrea	SD	Sudan
ES	Spain	SE	Sweden
ET	Ethiopia	SG	Singapore
FI	Finland	SH	Saint Helena
FJ	Fiji	SI	Slovenia
FK	Falkland Islands (Islas Malvinas)	SJ	Svalbard
FM	Micronesia	SK	Slovakia
FO	Faroe Islands	SL	Sierra Leone
FR	France	SM	San Marino
GA	Gabon	SN	Senegal
GB	United Kingdom	SO	Somalia
GD	Grenada	SO	Somaliland
GE	Abkhazia	SR	Suriname

GE	South Ossetia	ST	Sao Tome and Principe
GE	Georgia	SV	El Salvador
GF	French Guiana	SY	Syria
GG	Guernsey	SZ	Swaziland
GH	Ghana	TA	Tristan da Cunha
GI	Gibraltar	TC	Turks and Caicos Islands
GL	Greenland	TD	Chad
GM	Gambia, The	TF	French Southern and Antarctic Lands
GN	Guinea	TG	Togo
GP	Saint Martin	TH	Thailand
GP	Guadeloupe	TJ	Tajikistan
GP	Saint Barthelemy	TK	Tokelau
GQ	Equatorial Guinea	TL	Timor-Leste (East Timor)
GR	Greece	TM	Turkmenistan
GS	South Georgia & South Sandwich Islands	TN	Tunisia
GT	Guatemala	TO	Tonga
GU	Guam	TR	Turkey
GW	Guinea-Bissau	TT	Trinidad and Tobago
GY	Guyana	TV	Tuvalu
HK	Hong Kong	TW	China, Republic of (Taiwan)
HM	Heard Island and McDonald Islands	TZ	Tanzania
HN	Honduras	UA	Ukraine
HR	Croatia	UG	Uganda
HT	Haiti	UM	Palmyra Atoll
HU	Hungary	UM	Wake Island
ID	Indonesia	UM	Navassa Island
IE	Ireland	UM	Midway Islands
IL	Israel	UM	Kingman Reef
IM	Isle of Man	UM	Jarvis Island
IN	India	UM	Howland Island
IO	British Indian Ocean Territory	UM	Baker Island
IQ	Iraq	UM	Johnston Atoll
IR	Iran	US	United States
IS	Iceland	UY	Uruguay
IT	Italy	UZ	Uzbekistan
JE	Jersey	VA	Vatican City
JM	Jamaica	VC	Saint Vincent and the Grenadines
JO	Jordan	VE	Venezuela
JP	Japan	VG	British Virgin Islands
KE	Kenya	VI	U.S. Virgin Islands
KG	Kyrgyzstan	VN	Vietnam
KH	Cambodia	VU	Vanuatu
KI	Kiribati	WF	Wallis and Futuna

KM	Comoros	WS	Samoa
KN	Saint Kitts and Nevis	YE	Yemen
KP	Korea, North	YT	Mayotte
KR	Korea, South	ZA	South Africa
KW	Kuwait	ZM	Zambia
KY	Cayman Islands	ZW	Zimbabwe
KZ	Kazakhstan		

# State List

The following is a state list as some registries have restrictions on the combination of country and state.

<b>Australia</b> Our system is designed to only allow Australian states and territories to be used in combination with each other	
<b>Country Code</b>	Australia
<b>Available</b>	<ul style="list-style-type: none"><li>• NSW</li><li>• VIC</li><li>• QLD</li><li>• TAS</li><li>• ACT</li><li>• SA</li><li>• WA</li><li>• NT</li></ul>

<b>All Other Country Restrictions</b> We place no other system wide restrictions on the possible states/territories that can be used
---

# Domain Registration Information (All Extensions)

Domain name registrations are complicated for some extensions. We've outlined a number of restrictions, requirements and general information for domain name registrations using our API.

## General Information

Domain name registrations where no name servers are provided will adhere to the default name servers option you have set for your account. You may set all domain registrations to be parked and display your custom branded parking page or use the default name servers provided.

The majority of registrations are live with the registry and will be visible immediately in your account for management. Extensions such as org.au, asn.au and id.au are manually approved due to the restrictions imposed by the governing body, auDA.

## .AU Domain Name Registrations

Our API has the included ability to lookup ABN, ACN and RBN numbers provided during registration to improve the turnaround time on the domain name registration being successful. Domain names, which fail the automated lookup, are submitted for manual approval to be reviewed by staff. Mentioned in the 'Register (.au specific)' function is the information required, such as company name, ABN/ACN/RBN, etc.

We estimate a turnaround time of these extensions of around 1 hour during business hours from the time they are submitted.

## All Other Extensions

In specific cases where eligibility requirements for extensions need to be 'approved' in order for the domain name registration to be submitted please utilise the 'specialConditionsAgree' boolean value when submitting the domain name registration through the *domainRegister* command. For a list of TLD strings that are subject to eligibility approval at time of registration please contact support through the Synergy Wholesale management system.

## .AU Domain Registration Information

The following information is required for .AU domain name registrations. There are examples of how to use the additional fields for .au domain name registrations contained on our examples

Registrant ID Type (registrantIDType)	Eligibility Type (eligibilityType)	Eligibility ID Type (eligibilityIDType)
ABN	Charity	ABN
ACN	Citizen/Resident	ACN
OTHER	Club	ACT BN
	Commercial Statutory Body	NSW BN
	Company	NT BN
	Incorporated Association	OTHER
	Industry Body	QLD BN
	Non-profit Organisation	SA BN
	Other	TAS BN
	Partnership	TM
	Pending TM Owner	VIC BN
	Political Party	WA BN
	Registered Business	
	Religious/Church Group	
	Sole Trader	
	Trade Union	
Trademark		

# .US Domain Registration Information

The following information is required for .US domain name registrations.

Both the `nexusCategory` and `appPurpose` must be provided with the registration or contact update of a .US domain name.

<code>nexusCategory</code>	Definition
C11	US Citizen
C12	Permanent Resident
C21	US Organisation
C31	Foreign organisation doing business in US  Note: If this <i>nexusCategory</i> is provided when performing a <i>domainRegisterUS</i> or <i>updateContact</i> the country supplied for the registrant will be appended to the end of the <i>nexusCategory</i> (eg: C31/AU) to signify the origin as per .US requirements.
C32	Foreign organisation with US office  Note: If this <i>nexusCategory</i> is provided when performing a <i>domainRegisterUS</i> or <i>updateContact</i> the country supplied for the registrant will be appended to the end of the <i>nexusCategory</i> (eg: C32/AU) to signify the origin as per .US requirements.

<code>appPurpose</code>	Definition
P1	Business for profit
P2	Nonprofit
P3	Personal
P4	Educational
P5	Governmental

## ICANN Whois Data Verification Statuses

The table below will assist in understanding the values that are returned as part of the icannStatus string when performing a 'Domain Information Query' (domainInfo) on a domain name record.

API Status	Description
Pending Verification	The Registrant contact data is pending verification. The Registrant still needs to click the specially encoded link that was sent to them via email in order to validate the Registrant WHOIS data.
Pending Suspension	The domain name has surpassed the allowed time for Registrant contact validation and the domain name is pending suspension (the clientHold status will be applied to the domain shortly).
Suspended	The domain name has surpassed the allowed time for Registrant contact validation and the clientHold status has been applied to the domain. As such, DNS resolution has been suspended and any webhosting or email services will be affected.
Verified	The Registrant WHOIS data has been validated and no further action is required (unless the contacts on the domain are updated, at which time the domain name will enter Pending Verification again; or a WDRP/expiration notice bounces back).
N/A	The particular domain name is not subject to the ICANN Whois Data Verification ruleset and no action is required for Registrant contact validation (eg: .AU extensions and many other ccTLDs).

# **API Error Messages**

The following contains a list of error messages that may be returned for certain commands and related as detailed in our [API Documentation](#)

# API Error Messages - Domain Specific

The error messages shown below will be returned on a successful or failed attempt to use a function in the API.

Error Message	Description
AVAILABLE	Domain is available to register
UNAVAILABLE	Domain is unavailable to register
OK	The command executed successfully
OK_TRANSFER_PENDING	Pending submission to the registry to begin the transfer process
OK_TRANSFER_PENDING_COMPLETION	Transfer is pending registry completion
OK_TRANSFER_PUSH_PENDING	UK Specific - Awaiting existing registrar to push domain to our UK tag
OK_TRANSFER_REJECTED	Transfer rejected by registrant, registrar
OK_TRANSFER_CANCELLED	Transfer process has been cancelled
OK_TRANSFER_WAITING_AUTH	Transfer is pending registrant approval to proceed
OK_TRANSFER_MANUAL_REVIEW	Manual review is required for this domain transfer
OK_TRANSFER_UNKNOWN	Unknown status of domain transfer. Contact support team
OK_NO_RENEWAL	Renewal not required for transfer to take place
OK_RENEWAL	Renewal required during domain transfer submission
ERR_RESELLER_AUTH_FAILED	ResellerID and/or API Key is missing/empty
ERR_LOGIN_FAILED	Unable to login to the wholesale system
ERR_NO_USABLE_TLDS	Unable to locate any usable TLDs for reseller
ERR_DAEMON_FAILED	Unable to communicate with the domain daemon
ERR_DBLOOKUP_FAILED	Unable to perform database lookup
ERR_DOMAINREGISTER_FAILED	Register failed - Refer to reason in the provided errorMessage returned
ERR_DOMAINREGISTERAU_FAILED	Register failed - Refer to reason in the provided errorMessage returned
ERR_DOMAINREGISTERUK_FAILED	Register failed - Refer to reason in the provided errorMessage returned
ERR_DOMAINRENEWREQUIRED_FAILED	Renewal Required failed - Refer to reason in the provided errorMessage returned
ERR_TRANSFERDOMAIN_FAILED	Transfer failed - Refer to reason in the provided errorMessage returned
ERR_RESENDTRANSFEREMAIL_FAILED	Resend Email failed - Refer to reason in the provided errorMessage returned
ERR_UPDATEDOMAINPASSWORD_FAILED	Update Password failed - Refer to reason in the provided errorMessage returned
ERR_UPDATENAMESERVERS_FAILED	Name Servers failed - Refer to reason in the provided errorMessage returned
ERR_INVALID_DNSCONFIGTYPE	The provided dns config is invalid for the extension or is missing
ERR_RESELLER_NOT_AUTHORIZED	Reseller is not authorized to access specified domain
ERR_BULK_INPROGRESS	Unable to modify domain as a bulk update is in progress
ERR_NO_NAMESERVERS	Name servers missing from submitted array to be updated
ERR_NO_REGISTRY	Unable to determine the registry for which domain is required for update
ERR_DOMAINLOCK_FAILED	Lock failed - Refer to reason in the provided errorMessage returned
ERR_UNLOCKDOMAIN_FAILED	Unlock failed - Refer to reason in the provided errorMessage returned
ERR_RENEWDOMAIN_FAILED	Renew failed - Refer to reason in the provided errorMessage returned
ERR_NO_TLD_OPTIONS	Extension options unable to be obtained for specified TLD provided
ERR_DOMAINRELEASEUK_FAILED	UK Release failed - Refer to reason in the provided errorMessage returned
ERR_DOMAINTRANSFERUK_FAILED	UK Transfer failed - Refer to reason in the provided errorMessage returned
ERR_INVALID_UK	Domain provided is not a valid .UK extension
ERR_DOMAIN_EXISTS	Domain already exists in wholesale system
ERR_TRANSFER_EXISTS	Transfer already exists for supplied domain name being submitted

ERR_RESUBMITFAILEDTRANSFER_FAILED	Resubmit Transfer failed - Refer to reason in the provided errorMessage returned
ERR_NO_USABLE_TLDS	Unable to obtain allowed reseller TLDs
ERR_DOMAININFO_FAILED	Info failed - Refer to reason in the provided errorMessage returned
ERR_DOMAINCHECK_FAILED	Check failed - Refer to reason in the provided errorMessage returned
ERR_INITIATEAUCOR_FAILED	Submission of .AU Change of Registrant Failed. Refer to errorMessage Returned

## API Error Messages - DNS Specific

The error messages shown below will be returned on a successful or failed attempt to use a function in the API.

Error Message	Description
OK	The command executed successfully
ERR_RESELLER_AUTH_FAILED	ResellerID and/or API Key is missing/empty
ERR_LOGIN_FAILED	Unable to login to the wholesale system
ERR_VAR_EMPTY	Required variable is missing from the request
ERR_DBUPDATE_FAILED	Unable to perform database update - Refer to provided errorMessage
ERR_DBLOOKUP_FAILED	Unable to perform database lookup - Refer to provided errorMessage
ERR_RESELLER_NOT_AUTHORIZED	Reseller isn't authorized to access this supplied DNS zone
ERR_DOMAIN_MISSING	DNS zone doesn't exist for the supplied domain name
ERR_DOMAIN_EXISTS	DNS zone already exists for supplied domain name

# **API Usage Examples**

The following contains examples and further data for complex and specialist commands that are referenced and detailed in our API Documentation

## Example - .AU Domain Registrations

The following examples show the registrant and eligibility fields that must be provided to the API function depending on the type or domain and registrant.

### Company with an ABN as the registrant (com.au/net.au)

```
[registrantName] => SYNERGY WHOLESale PTY LTD  
[registrantID] => 17 165 245 901  
[registrantIDType] => ABN  
[eligibilityType] => Company
```

### Company with an ACN as the registrant (com.au/net.au)

```
[registrantName] => SYNERGY WHOLESale PTY LTD  
[registrantID] => 165 245 901  
[registrantIDType] => ACN  
[eligibilityType] => Company
```

### ABN of a sole trader as the registrant (com.au/net.au)

```
[registrantName] => John Doe  
[registrantID] => 17 123 123 123  
[registrantIDType] => ABN  
[eligibilityType] => Sole Trader
```

### Individual entity with a registered business number as the registrant (com.au/net.au)

```
[registrantName] => John Joe  
[registrantID] =>  
[registrantIDType] =>  
[eligibilityType] => Registered Business  
[eligibilityName] => Domain Factor Inc  
[eligibilityID] => BN149582021  
[eligibilityIDType] => VIC BN
```

### Company entity with a registered business number as the registrant (com.au/net.au)

```
[registrantName] => John Joe  
[registrantID] =>  
[registrantIDType] =>  
[eligibilityType] => Registered Business  
[eligibilityName] => Domain Factor Inc  
[eligibilityID] => BN149582021  
[eligibilityIDType] => VIC BN
```

### Individual as the registrant of a .id.au

Note: The registration of a .id.au domain name must have the eligibilityType set as Citizen/Resident with no other fields sent to thefunction.

```
[registrantName] => John Joe  
[eligibilityType] => Citizen/Resident
```

## Example - Name Server Update

The following example shows a successful update of name servers to a domain name.

### Example of successful name server update

```
(
  [resellerID] => *****
  [apiKey] => *****
  [domainName] => exampledomain.com.au
  [dnsConfig] => 1
  [nameServers] => Array
    (
      [0] => ns1.test.net.au
      [1] => ns2.test.net.au
      [2] => ns3.test.net.au
    )
)
```

As the name server update was successfully registered you can expect the following response from the API

```
[status] => OK
[errorMessage] => Domain name servers have been updated
```

## Example - SSL Certificate Purchase

The following example is provided for purchasing an SSL certificate using PHP/SOAP. SOAP is required to be compiled with PHP for the following to execute.

### Example Certificate Purchase Request

```
$client = new SoapClient(NULL, array('location' => 'https://api.synergywholesale.com/?wsdl', 'uri' => ''));

$data = array (
    'resellerID' => '4839202093',
    'apiKey' => '5k3k2kfjvkvsdkwk4k7k5k3k',
    'csr' => 'CSR Snipped',
    'productID' => '55',
    'firstName' => 'John',
    'lastName' => 'Doe',
    'emailAddress' => 'john@doe.com',
    'address' => 'PO Box 119',
    'city' => 'Beaconsfield',
    'state' => 'VIC',
    'postCode' => '3807',
    'country' => 'AU',
    'phone' => '+61.390245343',
    'fax' => '+61.380806481');

$output = $client->SSL_purchaseSSLCertificate($data);
```

### Example Successful Certificate Purchase Response

```
(
    [status] => OK
    [certStatus] => PENDING
    [certID] => 6859302
    [commonName] => Obtained from the CSR
)
```

## Example - DNS Record ID Calculation

The following example shows how the record ID value is calculated using a sha256sum of the record content.

### Example of how to calculate a DNS Record ID

```
$value = implode('|', [
    'name' => 'exampledomainname.com.au', // or www.domain.com, etc,
    'type' => 'NS', // the record type, TXT, MX, SRV, A, AAAA, etc.
    'content' => 'ns1.nameserver.net.au', // The normalised record content. Normalised means non-canonical domains (without a trailing dot)-
]);
// $value = exampledomainname.com.au|NS|ns1.nameserver.net.au

$tid = hash('sha256', $value);
// $tid = 'b17cfe936f7bb5c2c3c14fe07d8c579881144cbdd5bf959db127d2f28fd9f49'
```

# **Microsoft 365**

(Not yet available through the API)

The following commands allow for acquisition and management of Microsoft 365 email hosting services.

# Microsoft 365 Workflow

This workflow outlines how to purchase Microsoft 365 Subscriptions. Before starting, ensure that you (and your clients) are familiar with the Microsoft Customer Agreement - <https://aka.ms/customeragreement>.

## 1. Create a Client

**Command:** subscriptionCreateClient

This requires that you supply all relevant client details. The successful response will contain an identifier for the client, which is required for the purchase request.

**Note:** The agreement parameter must be set to 1/true/yes to proceed with the client creation.

## 2. List Purchasables

**Command:** subscriptionListPurchasable

This returns all Microsoft 365 products available to purchase. The successful response will contain a product ID which is required for the purchase request.

## 3. Purchase Subscription

**Command:** subscriptionPurchase

To complete the purchase, you need to provide the client identifier along with your Synergy Wholesale ID and API key. You also need to provide the productId and quantity (amount of seats) within the subscriptionOrder array. Once you receive a successful response, you will be able to see your client along with their subscriptions within the Synergy Wholesale Management System.

# Create a Client

The following command allows you to create a new client.

**Command:** subscriptionCreateClient

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
firstName	Mandatory	Client's first name
lastName	Mandatory	Client's last name
email	Mandatory	Client's email address
phone	Mandatory	Phone number of the client Note: Please use international format. E.g. +61.390245383
address	Mandatory	Physical address of the client
address2	Optional	Additional line for physical address
suburb	Mandatory	City/suburb of the client
postcode	Mandatory	Postcode of the client
state	Mandatory	State of the client
country	Mandatory	Country code of the client e.g. AU
company	Optional	Client's company name
agreement	Mandatory	Microsoft Customer Agreement. Note: This parameter must be set to 1/true/yes in agreement with the Microsoft Customer Agreement ( <a href="https://aka.ms/customeragreement">https://aka.ms/customeragreement</a> ) to proceed with the client creation process.
password	Optional	Client password Note: This will set the initial password on the subscription - note the password must meet complexity requirements (minimum 8-12 characters, 2 special characters, 1 uppercase and 1 number). If this value is not set a secure random password will be generated automatically.
description	Optional	Personalised description of the client to help with identification purposes
domainPrefix	Optional	A personalised value used to access tenancy with Microsoft

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the successful or failed command
identifier	Integer	Identifier of the newly created client
domainPrefix	String	Value required to access tenancy with Microsoft

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Successfully created client
[identifier] => 123123
[domainPrefix] => ab1cd2ef3gH4iJK5
```

## Example of a failed execution of the function

```
[errorMessage] => The request contained invalid data: First name must not be empty, State must be valid.
[status] => ERR_CREATE_CLIENT
```

# Updating Client Details

The following command allows you to update an existing customer's details.

**Command:** subscriptionUpdateClient

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the client
firstName	Mandatory	Client's first name
lastName	Mandatory	Client's last name
company	Optional	Client's company name
email	Mandatory	Email address for the new hosting service
phone	Mandatory	Phone number of the client Note: Please use international format. E.g. +61.390245383
address	Mandatory	Physical address of the client
address2	Optional	Additional line for physical address of the client.
suburb	Mandatory	City/suburb of the client
postcode	Mandatory	Postcode of the client
state	Mandatory	State of the client
country	Mandatory	Country code of the client e.g. AU
description	Optional	Personalised description of the client to help with identification purposes

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the successful or failed command

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Successfully updated client
```

## Example of a failed execution of the function

```
[errorMessage] => Identifier is missing or empty  
[status] => ERR_INVALID_IDENTIFIER
```

# Getting Client Details

The following command allows you to list the details of a particular client.

**Command:** subscriptionGetClient

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	The 'identifier' field will accept the identifier number of the client

Response		
Variable	Data Type	Description
status	String	Status is returned to advise of outcome
errorMessage	String	Text message of the successful or failed command
firstName	String	Client's first name
lastName	String	Client's last name
email	String	Client's email address
phone	String	Phone number of the client Note: Please use international format. E.g. +61.390245383
address	String	Physical address of the client
address2	String	Additional line for physical address of the client.
suburb	String	City/suburb of the client
postcode	String	Postcode of the client
state	String	State of the client
country	String	Country code of the client e.g. AU
company	String	Client's company name
domainPrefix	String	Randomised value required to access tenancy with Microsoft
clientStatus	String	Status of the client
description	String	Personalised description of the client

## Example of successful execution of the function

```
Array
(
    [status] => OK
    [errorMessage] => Successfully retrieved client
    [firstname] => John
    [lastname] => Smith
    [email] => test@example.com
    [phone] => 0401101010
    [address] => 123 Example Rd
    [address2] =>
    [suburb] => Narre Warren
    [postcode] => 3805
    [state] => VIC
    [country] => AU
    [company] => Example Company
    [domainPrefix] => 123testprefix
    [clientStatus] => Active
    [description] => Example Description
)
```

**Example of a failed execution of the function**

[errorMessage] => Identifier is missing or empty  
[status] => ERR\_INVALID\_IDENTIFIER

# Listing all Clients

The following command allows you to list all clients and their details.

**Command:** subscriptionListClients

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
page	Optional	An integer specifying the page you would like to return

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command
clientsList	array	Contains the nested details for each client
clientId	string	ID of the client
name	string	Name of the client
company	string	Company name of the client
status	string	The client's current status
totalSubscriptions	string	Total amount of subscriptions under this client

Example of successful execution of the function
<pre>[status] =&gt; OK [errorMessage] =&gt; Successfully retrieved clients [clientsList] =&gt; Array (   [0] =&gt; Array   (     [clientId] =&gt; 123123     [name] =&gt; John Smith     [company] =&gt; Example Company     [status] =&gt; Active     [totalSubscriptions] =&gt; 0   )   [1] =&gt; Array   (     [clientId] =&gt; 321321     [name] =&gt; Jane Doe     [company] =&gt; Different Company     [status] =&gt; Active     [totalSubscriptions] =&gt; 3   ) )</pre>

Example of a failed execution of the function
<pre>[errorMessage] =&gt; Failed to retrieve clients [status] =&gt; ERR_NO_CLIENTS_FOUND</pre>

# Listing Subscriptions Available for Purchase

The following command allows you to list all available subscriptions for purchase and its details.

**Command:** subscriptionListPurchasable

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command
subscriptionList	array	Contains the nested details for each subscription
productId	string	ID of the product
productName	string	Name of the product
productDescription	string	Description of the product
productType	string	Product type
monthlyCost	decimal	Monthly cost of the subscription
maxSeats	string	Maximum amount of seats available for the product

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Successfully retrieved purchasable subscriptions and addons
[subscriptionList] => [0] => Array
(
  [0] => Array
  (
    [productId] => 321312
    [productName] => Microsoft 365 Business Basic
    [productDescription] => Microsoft 365
    [productType] => Subscription
    [monthlyCost] => 12.30
    [maxSeats] => 300
  )
  [1] => Array
  (
    [productId] => 321313
    [productName] => Microsoft 365 Business Premium
    [productDescription] => Microsoft 365
    [productType] => Subscription
    [monthlyCost] => 12.30
    [maxSeats] => 300
  )
)
```

## Example of a failed execution of the function

```
[errorMessage] => Unable to login to wholesale system
[status] => ERR_LOGIN_FAILED
```

# Purchase a Subscription/Addon

The following command allows you to purchase one or more subscriptions. This command is also used for purchasing addons.

**Command:** subscriptionPurchase

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the client
subscriptionOrder	Mandatory	Array containing the product ID and quantity of the subscription/addon being purchased
productId	Mandatory	ID of the subscription you wish to purchase Note: This can be obtained through the subscriptionListPurchasable endpoint
quantity	Mandatory	The amount of seats you wish to purchase for this subscription

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command
subscriptionList	array	Contains the nested details for each subscription
subscriptionId	string	ID of the subscription
productId	string	ID of the product
productName	string	Name of the product
productDescription	string	Description of the product
productType	string	Product type
monthlyCost	decimal	Monthly cost of the subscription
quantity	string	Amount of seats
status	string	The subscriptions current status

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Successfully completed purchase
[subscriptionList] => Array
(
  [0] => Array
  (
    [subscriptionId] => 123123
    [productId] => 321312
    [productName] => Microsoft 365 Business Basic
    [productDescription] => Microsoft 365
    [productType] => Subscription
    [monthlyCost] => 12.30
    [quantity] => 1
    [status] => Pending
  )
)
```

## Example of a failed execution of the function

```
[errorMessage] => No products specified in order
[status] => ERR_ORDER_EMPTY
```

# Get Subscription Details

This command allows you to retrieve details relating to a subscription.

**Command:** subscriptionGetDetails

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the subscription

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command
productId	string	ID of the product
productName	string	Name of the product
quantity	string	Amount of seats
monthlyCost	decimal	Monthly cost of the subscription
tasks	array	Tasks currently pending execution on the subscription
action	string	Name of the pending task
status	string	Status advising of the outcome of the task when executed
last_updated	dateTime	Time of when the task was last actioned
subscriptionStatus	string	Status of the subscription

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Successfully retrieved subscription
[productId] => 213213
[productName] => Microsoft 365 Business Basic
[quantity] => 1
[monthlyCost] => 12.30
[tasks] => Array
(
  [0] => Array
  (
    [action] => CREATE_SUBSCRIPTION_ORDER
    [status] => Success
    [last_updated] => 2022-05-18 23:58:44
  )
)
[subscriptionStatus] => Active
```

## Example of a failed execution of the function

```
[errorMessage] => Unable to retrieve subscription details
[status] => ERR_SUBSCRIPTION_NOT_FOUND
```

# Update Subscription Quantity

This command allows you to update the quantity/seats of your subscriptions.

**Command:** subscriptionUpdateQuantity

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the subscription
quantity	Mandatory	Amount of subscriptions you wish to change to Note: This will replace the current amount of subscriptions you have e.g. You have 3, you send the quantity through as 1, you will now have 1 subscription

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command
newMonthlyCost	string	New monthly cost of subscription after quantity update
cost	string	The pro rata cost of the quantity update

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Updated seats from: 1 to 2  
[newMonthlyCost] => 24.53  
[monthlyCost] => 12.26
```

## Example of a failed execution of the function

```
[errorMessage] => Unable to retrieve subscription details  
[status] => ERR_SUBSCRIPTION_NOT_FOUND
```

# Suspend Subscription

This command allows you to suspend a subscription.

**Note:** If a subscription remains in a suspended state for 90 days or more from the subscription term end date, the subscription will be permanently deleted.

**Command:** subscriptionSuspend

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the subscription

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command

Example of successful execution of the function
[status] => OK [errorMessage] => Successfully suspended subscription

Example of a failed execution of the function
[errorMessage] => ERR_UPDATE_FAILED [status] => Failed to update the subscription

# Unsuspend Subscription

This command allows you to unsuspend a subscription.

**Command:** subscriptionUnsuspend

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the subscription

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command

## Example of successful execution of the function

```
[status] => OK  
[errorMessage] => Successfully unsuspended subscription
```

## Example of a failed execution of the function

```
[errorMessage] => ERR_UPDATE_FAILED  
[status] => Failed to update the subscription
```

# List Client Subscriptions

This command allows you to list all of a client's subscriptions.

**Command:** `subscriptionListClientSubscriptions`

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the client

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command
subscriptionList	array	Contains the nested details for each subscription
subscriptionId	string	ID of the subscription
productId	string	ID of the product
productName	string	Name of the product
productDescription	string	Description of the product
productType	string	Product type
monthlyCost	decimal	The cost of the product monthly
quantity	string	Amount of seats
subscriptionStatus	string	Status of the subscription

## Example of successful execution of the function

```
[status] => OK
[errorMessage] => Successfully retrieved subscriptions
[subscriptionList] => Array
(
  [0] => Array
  (
    [subscriptionId] => 123123
    [productId] => 321312
    [productName] => Microsoft 365 Business Basic
    [productDescription] => Microsoft 365
    [productType] => Subscription
    [monthlyCost] => 12.30
    [quantity] => 1
    [status] => Active
  )
  [1] => Array
  (
    [subscriptionId] => 333123
    [productId] => 321313
    [productName] => Microsoft 365 Business Premium
    [productDescription] => Microsoft 365
    [productType] => Subscription
    [monthlyCost] => 32.10
    [quantity] => 3
    [status] => Active
  )
)
```

**Example of a failed execution of the function**

[errorMessage] => ERR\_NO\_SUBSCRIPTIONS\_FOUND  
[status] => Failed to retrieve subscriptions

# Terminate Subscription

This command allows you to terminate a subscription.

**Command:** subscriptionTerminate

Request		
Variable	Required	Description
resellerID	Mandatory	Reseller ID provided at account confirmation
apiKey	Mandatory	API Key can be obtained from the wholesale system interface
identifier	Mandatory	Identifier of the subscription

Response		
Variable	Data Type	Description
status	string	Status is returned to advise of outcome
errorMessage	string	Text message of the successful or failed command

Example of successful execution of the function
[status] => OK [errorMessage] => Successfully terminated subscription

Example of a failed execution of the function
[errorMessage] => ERR_UPDATE_FAILED [status] => Failed to terminate the subscription