

## CDYNE Email Verification Web Service

Version 2.1

**WSDL:** <http://ws.cdyne.com/emailverify/Emailvernotestemail.asmx?wsdl>

**Testing URL:** <http://ws.cdyne.com/emailverify/emailvernotestemail.asmx>

**Developer Resources:** [http://wiki.cdyne.com/wiki/index.php?title=Email\\_Verification](http://wiki.cdyne.com/wiki/index.php?title=Email_Verification)

**Product Web Page:** <http://www.cdyne.com/products/email-verification.aspx>

**Description:** Utilize this XML Web Service to verify any email address. CDYNE Email Verification is designed to check the deliverability of email addresses. This service can be implemented in any system or application connected to the Internet and is designed to be a quick and non-intrusive method to validate an email address without spamming.

Companies that rely on email communications use the service to keep their lists of addresses corrected and to flag those addresses that are no longer deliverable. CDYNE Email Verification is quick enough to perform entry-point checking for online applications or data collection sites.

Email Verification is part of CDYNE's suite of fraud detection products that include Postal Address Verification, Phone Verification, Phone Notify!, and Death Index for complete customer record validation.

**Required License Information:** The *LicenseKey* parameter that is seen in most operations of this Web Service is required to authenticate users invoking this Web Service. You must obtain a valid License Key. You can sign up for this service by using the 'Buy Now' button located on the Product Web Page (see link at top of this page) or by contacting CDYNE for a limited Trial License Key.

*EXAMPLE:*

Licensing parameter seen in numerous operations	Sample Input
LicenseKey	F01d89fd-5155-5455-5585-e84ab8de8591

### Operations:

- AdvancedVerifyEmail
- ReturnCodes
- VerifyEmail
- VerifyMXRecord

▼ **AdvancedVerifyEmail**

<b>Operation: AdvancedVerifyEmail Input</b>		
Parameter Name:	Data Type	Sample Input
<b>Email:</b> Input email address to verify.	<i>String</i>	info@cdyne.com
<b>Timeout:</b> Input # of seconds to timeout verification process.	<i>Integer</i>	10
<b>LicenseKey:</b> See page 1 for more info.	<i>String</i>	F01d89fd-5155-5455-5585-e84ab8de8591

<b>Operation: AdvancedVerifyEmail Output</b>		
Parameter Name:	Data Type	Sample Output
<b>ResponseText:</b> Returns response in text.	<i>String</i>	Mail Server will accept email
<b>ResponseCode:</b> Returns response code: 0 – Invalid Email address (bad email) 1 – Not Used (good email) 2 – Verified Email address (good email) 3 – Mail Server will accept email (good email) 4 – User Not Found (bad email) 5 – Email Domain Not Found (bad email) 6 – Not Used (good email) 7 – SMTP/Timeout error (good email) 8 – Domain valid. Email server down (good email) 9 – License Key Allowance exceeded		3
<b>LastMailServer:</b> Returns last mail server.	<i>String</i>	Mail.cdyne.com
<b>GoodEmail:</b> Returns true if email address is good or false if it is not.	<i>String</i>	True

▼ **ReturnCodes**

- There is no input value for this operation.
- Returns all response codes and text.

**Operation: ReturnCodes Output**

**ResponseCode:** Returns response code:  
 0 – Invalid Email address (bad email)  
 1 – Not Used (good email)  
 2 – Verified Email address (good email)  
 3 – Mail Server will accept email (good email)  
 4 – User Not Found (bad email)  
 5 – Email Domain Not Found (bad email)  
 6 – Not Used (good email)  
 7 – SMTP/Timeout error (good email)  
 8 – Domain valid. Email server down (good email)  
 9 – License Key Allowance exceeded

▼ **VerifyEmail**

-Does the same operation as AdvancedVerifyEmail, except timeout is automatically set to 5 seconds.

**Operation: VerifyEmail Input**

Parameter Name:	Data Type	Sample Input
<b>Email:</b> Input email address to verify.	<i>String</i>	info@cdyne.com
<b>LicenseKey:</b> See page 1 for more info.	<i>String</i>	F01d89fd-5155-5455-5585-e84ab8de8591

**Operation: VerifyEmail Output**

Parameter Name:	Data Type	Sample Output
<b>ResponseText:</b> Returns response in text.	<i>String</i>	Mail Server will accept email
<b>ResponseCode:</b> Returns response code: 0 – Invalid Email address (bad email) 1 – Not Used (good email) 2 – Verified Email address (good email) 3 – Mail Server will accept email (good email) 4 – User Not Found (bad email) 5 – Email Domain Not Found (bad email) 6 – Not Used (good email) 7 – SMTP/Timeout error (good email)		3

8 – Domain valid. Email server down (good email) 9 – License Key Allowance exceeded		
<b>LastMailServer:</b> Returns last mail server.	<i>String</i>	Mail.cdyne.com
<b>GoodEmail:</b> Returns true if the email address is good or false if it is not.	<i>String</i>	True

▼ **VerifyMXRecord**

<b>Operation: VerifyMXRecord Input</b>		
Parameter Name:	Data Type	Sample Input
<b>Email:</b> Input email address to verify. <i>String</i>		info@cdyne.com
<b>LicenseKey:</b> See page 1 for more info. <i>String</i>		F01d89fd-5155-5455-5585-e84ab8de8591

<b>Operation: VerifyMXRecord Output</b>		
Parameter Name:	Data Type	Sample Output
<b>int:</b> Returns code for verification. <i>0 – Email Domain Invalid</i> <i>-9999 – Try again Later</i> <i>All others – Email Domain Valid</i>	<i>Integer</i>	1