CYREN
IP Reputation Daemon (ctIPd)
Integration Manual
v4.0

ctIPd Module and Documentation Usage Restrictions

This program and the accompanied documentation is SECRET AND CONFIDENTIAL, and constitute a proprietary trade secret of CYREN Inc. (herein after referred to as "CYREN").

No person is allowed to copy, decompile, reverse engineer, use, sublicense or otherwise access this program unless the prior express, written consent is received from CYREN. The possession and use of this program shall be governed by the terms of a license agreement between CYREN and each authorized licensee. Unauthorized use of this program is strictly prohibited, and those perpetrating such unauthorized uses shall be prosecuted to the fullest extent of the law. The confidentiality and non-disclosure obligations of licensee shall be strictly maintained at all times by licensee and licensee, in receiving a copy of this program, acknowledges that it shall not be disclosed to third parties; rather, only to employees or consultants having a firm need to know, and provided that they are bound by confidentiality restrictions at least as restrictive as those adopted by licensee within the framework of its relationship with CYREN.

The failure to maintain confidentiality will likely cause severe damages and irreparable harm to CYREN and, therefore, in addition to any other remedies and rights available at law, CYREN shall be entitled to seek injunctive relief without the need for the posting of any bond or other guarantee.

Contacts

Any technical questions you or your developers have about using the ctwsd should be addressed to support@cyren.com.

Trademark and Copyright Statement

ctIPd is a licensed SDK product featuring patented technology. CYREN's patented solution is protected by U.S. patent #6,330,590.

RPD, Zero-Hour Protection, IPRep, ctIPd are trademarks of CYREN Inc. For more information, visit our website: http://www.cyren.com/

Linux is a trademark of Linus Torvalds. FreeBSD is a registered trademark of Wind River Systems, Inc. RBL is a trademark for the proprietary MAPS DNSBL. All other trademarks and registered trademarks are the property of their respective owners.

About CYREN

CYREN^{TM} provides proven Internet security technology to more than 150 security companies and service providers including 1&1, Check Point, F-Secure, Google, Microsoft, Panda Security, Rackspace, US Internet, and WatchGuard, for integration into their solutions. CYREN's GlobalView™ and patented Recurrent Pattern Detection™ (RPD™) technologies are founded on a unique cloud-based approach, and protect effectively in all languages and formats. CYREN Antivirus utilizes a multi-layered approach to provide award winning malware detection and industry-leading performance.

CYREN technology automatically analyzes billions of Internet transactions in real-time in its global data centers to identify new threats as they are initiated, enabling our partners to protect end-users from spam and malware, and ensure safe, compliant browsing. The company's expertise in building efficient, mass-scale security services mitigate Internet threats for thousands of organizations and hundreds of millions of users in 190 countries.

CYREN, formerly known as Commtouch, was founded in 1991, is headquartered in the US in McLean, Virginia, with offices in Palo Alto, California, Herzliya, Israel, Berlin, Germany, and Reykjavik, Iceland.

For more information about enhancing security offerings with CYREN technology, visit our website at www.cyren.com, see our blog at http://blog.cyren.com or write to support@cyren.com.

Table of Contents

1 Introduction

CYREN GlobalView™ Mail Reputation Service, provided by the CYREN IP Reputation daemon (ctIPd) is used primarily to weed out spam messages and email-borne malware at the entry point, before these messages enter the customer's messaging network, thereby relieving the need for resource-consuming downstream filtering. By applying Mail Reputation services to senders of inbound messages before or during the SMTP connection and before they enter the messaging network, ctIPd delivers cost-effective benefits such as:

- Reducing IT resources (for example server count, CPU load, storage, etc.)
- Eliminating multiple security risks
- Reducing the level of false positives
- Minimizing the cost of downstream filtering
- Lowering the overall bandwidth consumption
- Optimizing IT labor required to manage the overall messaging process

CYREN GlobalView Mail Reputation services can be used as part of an overall strategy to optimize network accessibility. By weeding out these resource-draining SMTP connections before any actual messagesare transmitted, CYREN's solution enables the network's messaging processes to work more efficiently, so that valuable IT resources are correctly focused on allowing legitimate sources full and uninterruptedaccess while making access to unauthorized sources that are attempting to abuse the network, more difficultto achieve.

Many public services nowadays already deliver free information about blacklisted sources; typically, these lists are static with limited commitment for accuracy by the owners. Other commercial services also enable setting central as well as personal policies for whitelisted sources during the SMTP connection. However, these solutions are not designed to handle the vast 'grey' area of sources for which little or no information is readily available. Included in this group, for example, are millions of machines hijacked on a daily basis and forced into botnets and armies of zombies discrediting their reputation, dynamically. Many of these bots are detected at a later time and are removed and thus the reputation of the once-infected machines is raised.

CYREN analyzes hundreds of millions of messages every day. Included in this analysis is a process whereby CYREN dynamically and in real-time classifies and reclassifies the reputation of each source IP with a series of meaningful and measurable values. These values are constantly updated to include the most up-to-date information and reflect even the slightest change incredibility.

ctIPd™ is an embedded reputation engine with a small footprint responsible for maintaining communication with the CYREN Datacenter. ctIPd delivers reputation data in real-time to messaging, security and networking devices. These devices are responsible for posting queries to ctIPd over HTTP, UDP, or RBL/RBL+ protocol requesting reputation data on source IP addresses attempting to establish SMTP connectionsfor sending messages to recipients.

ctIPd analyzes all available data and provides a suggested call-to-action to the querying device. In response to ctIPd's returned call-to-action, the querying device will accept, tempfail, or permfail the connection from a sending SMTP source.

Alternatively, a non-CYREN flow control or connection management application on the client-side can be configured to receive reputation data from ctIPd. It can then weigh the responses and make its own determination for the appropriate actions to implement.

By applying ctIPd-recommended actions based on real-time, precise information about globally monitored behavior of source IP(s), the customer is able to throttle unknown and suspected sources while freeing bandwidth to offer free access to desired business contacts.

This document presents a detailed overview of the CYREN GlobalView Mail Reputation services, focusing on ctIPd, the various interfacing protocols, its deployment options, installation process, configuration settings and its overall functionality. It focuses on how to integrate ctIPd into the application and the information you need to accomplish this. Because ctIPd can be implemented in a variety of ways according to the needs and preferences of each application, a separate document, *ctIPD Implementation Guidelines*, is also available.

1.1 Components of the GlobalView Mail Reputation Services

The GlobalView Mail Reputation services involve the following components:

- Querying device
- ctIPd daemon
- ctIPd protocol
- CYREN Datacenter

Querying Device

For the purposes of this document, the term "querying device" is used as a generic term for mail transfer agents (MTA), security appliances, networking devices, or any device that is capable of receiving email messages or monitoring SMTP traffic and generating a query to ctIPd over HTTP, UDP, or RBL/RBL+ protocols. Once a response from ctIPd is received, the querying device is responsible for applying connection management decisions and flow control actions based on ctIPd's response.

ctIPd

The CYREN IP Reputation embedded daemon (ctIPd) performs various functions, from receiving and processing incoming requests from querying devices to determining the IP reputation of specific sources and quickly responding to the querying devices with details on several key data types along with recommended action. Typically, ctIPd is deployed on-site in order to guarantee high performance and availability to local querying devices.

ctIPd Protocol

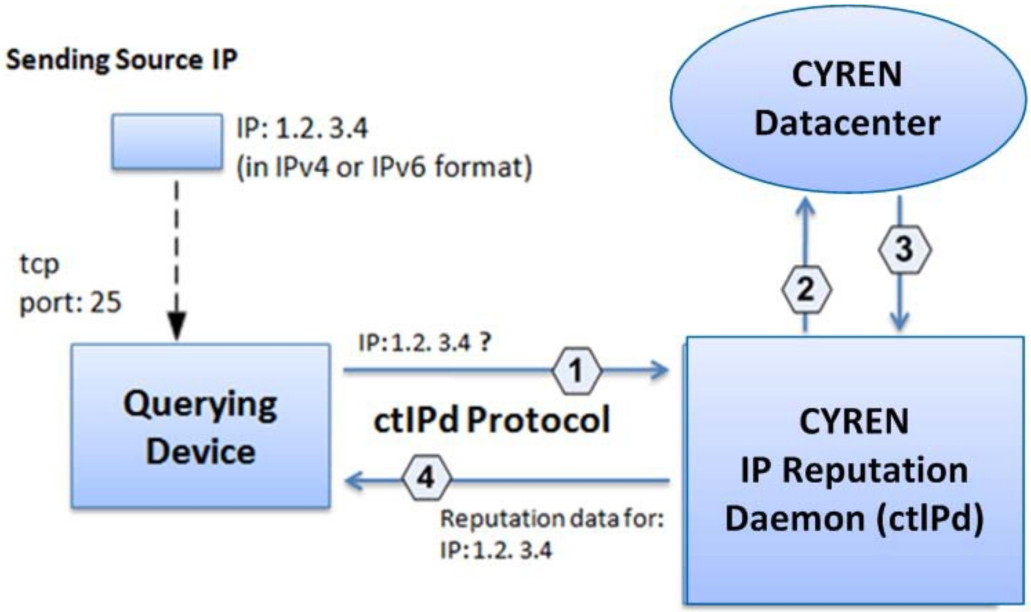
In order to enable communication between a querying device and ctIPd and easy integration by its OEM partners, CYREN has developed a simple communication protocol that is detailed in this document. This protocol enables OEM partners to communicate with ctIPd and thereby to provide IP reputation services to their users. Communication between ctIPd and the querying device can be accomplished over HTTP, UDP or RBL/RBL+ interfaces. For more information, see ctIPd Protocol.\

CYREN Datacenter

The CYREN Datacenter monitors global email traffic in real-time (24*7*365) from various sources on an ongoing basis and maintains a vast database of IP reputation classifications that are determined based on numerous dynamically changing parameters.

1.2 System Architecture and Data Flow

Although the data flow of ctIPd can vary depending on configuration settings and deployment scenarios, a typical data flow is detailed below:



System Architecture and Data Flow

1. A source IP attempts to send messages to recipients. Typically, this would initiate an SMTP connection to transfer the data. Before any data is transferred between the sending IP and the mail server, the querying device uses the CYREN ctIPd protocol to generate a query to ctIPd about the reputation of the sending IP.
2. ctIPd prepares and forwards a query to a CYREN Datacenter to retrieve the most up-to-date information based on global and dynamic behavior of the source IP.
3. The CYREN Datacenter responds to ctIPd with current information regarding the sender IP.
4. ctIPd collects IP reputation data along with some cumulated stats about local exposure to the same sender.
5. After analyzing all available data and formulating a call-to-action, ctIPd reformats the response into the appropriate protocol and sends the response back to the querying device over the same protocol used by the initial query.

The querying device, upon receiving the response from ctIPd, implements ctIPd's recommended call-to-action. Alternatively, ctIPd can be configured to forward raw IP reputation data to the querying device with or without the call-to-action.

The querying device can also separately initiate passing a series of IP addresses to ctIPd that are already known to it as being 'good' and 'bad' sources as well as passing the decision that it has made on the queried IP after the response from ctIPd was received. This information may have been gained by another querying device, policy, or reputation service. When passed to CYREN, it can become part of the database of IP reputation classifications. This will assist ctIPd in quickly and correctly responding to future queries about these sources.

1.3 Local Cache Options

To improve speed and efficiency in providing IP reputation data, ctIPd has two local cache options for storing and using data effectively:

- Raw Reputation cache
- Decision Manager cache

In both cases, these caches are regularly and automatically updated by the CYREN Datacenter, as explained in the following sections.

1.3.1 Raw Reputation Cache

The Raw Reputation cache maintains a local cache of all IP reputation data sent to ctIPd by the Datacenter in response to ctIPd queries. As new queries are received, ctIPd checks the local Raw Reputation cache to see if data from there can be reused.

1.3.2 Decision Manager Cache

The local Decision Manager cache is a separate cache that contains statistics on already-queried IP addresses. It is used to compute dynamically-changed recommended actions (throttling) each time the sender attempts to establish a new SMTP connection. This cache is applicable only when the Decision Manager functionality is enabled, which is the default operational mode for ctIPd.

1.4 ctIPd Package Contents

The ctIPd package contains the following:

- ctIPd daemon and associated binary files
- ctIPd documentation
- Sample files for quick evaluation and testing

1.5 Internal Directory Structure

After unpacking, the ctIPd internal directory structure consists of the following:

./ctipd-<version>/

bin	Binary files and configuration files
bin/reports	Binary files needed for generating reports
bin/snmp	SNMP files
snmp/mibs	SNMP MIB files
docs	ctIPd documentation
samples	Sample scripts for evaluation and testing

1.6 Additional Resources

In addition to this document, the following additional resources are available within the ctIPd package:

- *ctIPd Product Description*
- *ctIPd Implementation Guidelines*
- *ctIPd Reports*
- *ctIPd Testing and Evaluation Guidelines*

2 ctIPd Configuration

A default configuration file is provided with the ctIPd package containing the necessary configuration parameters for the daemon. The administrator can configure a single ctIPd configuration file and then copy it to other locations to be used by other daemons, but each instance of a CYREN daemon uses its own unique ctIPd configuration file.

By default, the name of the file is ctipd.conf and is located in the same directory as the ctIPd binary. You can rename this file. You can also place it in a different location by specifying the new name and location in the command line while running ctIPd.

Most of the parameters in the configuration file are optional, and have default values that are applied if a different value is not specified. However, in order for connectivity with the CYREN Datacenter to be established, valid ServerAddress= and LicenseKey= values must be entered. Additionally, if you plan to use the RBL/RBL+ interface, specify the DomainName= in use for the ctIPd host.

Note: *If you change the configuration file while ctIPd is operating, you must restart ctIPd in order to have these changes take effect. ctIPd only checks the configuration file at startup and does not Update the current configuration parameters until the next startup.*

2.1 Sample Configuration File

Following is a copy of the default configuration file. Where a default value is set, it appears below as well. The next section provides a detailed description for each parameter.

```
[General]
# User name associated with ctIPd
# RunAsUser = ctipd

# Group associated with ctIPd
# RunAsGroup = ctipd

# Defines whether ctIPd is deployed over LAN ‘0’ or over WAN ‘1’
# UseAuthMode = 0

[IPRep]
# Server address at CYREN Datacenter
ServerAddress =
# License key code for ctIPd
LicenseKey =
# Maximum number of records stored in the local cache
# CacheMaxRecords = 100000

# Cache file name
# CacheFileName = /tmp/ctipd.cache

# Periodic save interval in minutes, use ‘0’ to disable periodic save
# CacheSaveInterval = 60

# Maximum concurrent requests
# RequestConcurrency = 3000

# Specifies whether raw reputation data results are included in the response
# SendRawData = 0

#   If you connect to the Internet through a proxy server, you
#   should uncomment the following parameters and assign appropriate
#   values.

# ProxyPort = 80
# ProxyServerAddress = myproxy
# ProxyAuth = NoAuth
# ProxyUserName = user@proxy
# ProxyUserDomain = domain.com
# ProxyPassword = 1234
# ProxyAccess = 0

[HTTP]

#  This section enables you to specify the TCP port on the daemon to which
the #  client connects, and the relevant connectivity and performance
parameters.

# Indicates whether interfacing over HTTP protocol is enabled
# Enabled = yes

# Port name/number for HTTP server
# Port = 8080

# Maximum number of HTTP threads
# MaxThreads = 100

# Threads to create on startup
# InitialThreads = 0

# Receive timeout from HTTP client
# ReceiveTimeout = 5000

# Size of TCP backlog queue
# ListenBackLog = 50

# IP Address to bind to the socket
# BindingAddress = INADDR_ANY

[UDP]

# This section enables you to specify the UDP server port on the daemon to
# which the client connects, and the relevant connectivity and performance
# parameters.

# Indicates whether interfacing over UDP protocol is enabled
# Enabled = yes

# Port name/number for UDP server
# Port = 5678

# Maximum number of UDP threads
# MaxThreads = 100

# Threads to create on startup
# InitialThreads = 0

# Receive buffer size [bytes]
# ReceiveBuff = 65535

# Size of waiting requests queue
# MaxQueueSize = 3000

# Age threshold for discarding non-handled requests in the queue
# RequestTimeout = 30

# IP Address to bind to the socket
# BindingAddress = INADDR_ANY

[RBL]
# The section enables you to specify the RBL server port on the daemon to
# which the client connects, and the relevant connectivity and performance
# parameters.

# Indicates whether interfacing over RBL protocol is enabled
# Enabled = no

# If enabled, specify the domain name in use for the ctIPd host
DomainName =

# Port name/number for RBL server
# Port = 53

# Maximum number of RBL threads
# MaxThreads = 100

# Threads to create on startup
# InitialThreads = 0

# Receive buffer size [bytes]
# ReceiveBuff = 65535

# Size of waiting requests queue
# MaxQueueSize = 3000

# Age threshold for discarding non-handled requests in the queue
# RequestTimeout = 30

# IP Address to bind to the socket
# If BindingAddress is empty (or commented), BindingAddress is set to
# INADDR_ANY.
# BindingAddress =

# Define the Start of Authority (SOA) resource record.
# Start of Authority (SOA) resource record
# SOA = DomainName=; MailBoxDomainName=; Serial=; Refresh=; Retry=; Expire=;
# Minimum=;

# Enables you to define the Name Server.
# NS =

# Name of the files containing information on rejected messages
# TXT_TempfailFile = ./tempfail.txt
# TXT_PermfailFile = ./permfail.txt
```

```
[Stats]
# Indicates whether ctIPd will maintain statistics.
# Enabled = yes

# Port name/number for stats server. For Windows, use a numeric value only.
# Port = /tmp/ctipd.stats

# IP Address to bind to the socket
# If BindingAddress is empty (or commented), BindingAddress is set to
# INADDR_ANY.
# BindingAddress = INADDR_ANY

[DecisionManager]
# Indicates whether ctIPd's internal decision manager is enabled
# Enabled = yes

# MaxRecords = 100000

[LogServer]
# The method used for logging. By default, value is ‘0’, meaning
# no logging is performed.

# LogMethod = 0

#    If you change the LogMethod to ‘1’, you can create a directory
#    in which to store the logging files and then specify the name of the
#    directory for the DirName parameter, or use the default value.

# If enabled, specify the domain name in use for the ctIPd host
# DirName =/tmp/ctipd_log

# Daemon name for which logging files are maintained.
# DaemonName = 0001

# Port via which logging data is transmitted
# Port = 23456

# Maximum combined storage size allocated for all logfiles
# MaxSize = 2000

# Maximum number of hours for which records are kept
# MaxPeriod = 24
# Maximum number of file size before new logfile is started
# MaxFileSize = 50

# IP Address to bind to the socket
# BindingAddress = INADDR_ANY
```

2.1.1 [General]

The [General] section enables you to define the user and group under which ctIPd daemon is running. This is primarily used to grant different levels of permissions.

RunAsUser = ctipd

The `RunAsUser` option enables you to define the user name associated with ctIPd.

RunAsGroup = ctipd

The `RunAsGroup` option enables you to define the group associated with ctIPd.

UseAuthMode = 0

The `UseAuthMode` option defines whether ctIPd is deployed over LAN '0' or over WAN '1'. For more information, review the Deployment Options section in the *ctIPd Implementation Guide*. If ctIPd is deployed over WAN you need to ensure that HTTP requests are sent with the X-CTCH-Key header as explained later in the section ctIPd Protocol. Default value: `0` (disabled).

2.1.2 [IPRep]

The [IPRep] section enables you to specify ctIPd-related options for:

- Establishing communication with a CYREN Datacenter.
- Configuring the local cache to expedite response time. The cache is also saved to a file on a regular basis so that, upon restart, ctIPd can reload the most recently saved file as an initial cache base. The file name, location, and save time interval are all configurable.
- Setting the maximum number of concurrent threads.

ServerAddress = <DNS string>

The DNS string is the server address at CYREN Datacenter. Contact your CYREN sales representative to obtain a valid DNS string that will uniquely identify your queries. This is a mandatory field.

LicenseKey = <license key code>

Enter the license key code for ctIPd. If an incorrect key code is entered, the CYREN Datacenter will not be able to authenticate the organization and therefore will not provide IP reputation services. Contact your CYREN sales representative to obtain a valid key code. This mandatory value is supplied as a parameter in the Connection String, and consists of the following:

- **CYREN token:** 20-character unique identifier provided by CYREN to identify the OEM partner or Service Provider
- **OEM token:** A unique identifier (up to 35 alphanumeric characters) provided by the OEM partner or Service Provider

The OEM/Service Provider identifier should distinguish between each user, device, or installation. In cases where more than one CYREN product or service is installed on the same device, a unique token should be created per instance. The token should be unique for the lifetime of the host application and should not be changed so that the same OEM/Service Provider token is used each time the application is initiated.

It can be based on hardware or software-specific data. CYREN needs this full license key format to offer the highest level of customer support and service. The format for this concatenated parameter uses a colon delimiter, as follows:

```
LicenseKey=<CYREN token>:<unique OEM token>
```

Example:

```
LicenseKey=0001K032B1010W167E2B:12345-1234A-55555
```

CacheMaxRecords = <value>

The maximum size of the local IP reputation cache is defined in the `CacheMaxRecords` option. Once the cache reaches the specified amount, the oldest records are overwritten by newer ones, thus limiting the cache records to the specified value set in the configuration file. The default value is: `100,000`.

CacheFileName

The `CacheFileName` option specifies the name and path of the file replicating the local cache data. By default, this is: `/tmp/ctipd.cache`.

CacheSaveInterval

The `CacheSaveInterval` option specifies the amount of time, in minutes, that the local IP reputation cache is saved to a file. By default the interval, in minutes, is: `60`. To disable the automatic save functionality, enter a value: `0` (zero)

RequestConcurrency

The `RequestConcurrency` option enables you to specify the maximum number of concurrent requests that can be processed by ctIPd at any given time. Any amount above this value will be held temporarily in the queue until the daemon is available to process new requests. In the case of HTTP, connections may time out. In the case of UDP, the value of `MaxQueueSize` is applied (`3000`). The default value of `3,000` requests is extremely high and is unlikely to be reached. You can change the value as desired.

SendRawData

The `SendRawData` option specifies whether raw reputation data files are included in the response sent back to the querying device. This option does not apply to RBL/RBL+. By definition, the value is `0`.

Proxy Settings

If you connect to the Internet through a proxy server, the following settings can be assigned appropriate values. You will also need to remove the comment (#) before these values become active.

ProxyPort	The proxy port through which the ctIPd can communicate with the Internet.
ProxyServerAddress	The proxy server address.

ProxyAuth The authentication method of the proxy server. Options include: NTLM, Basic, NoAuth. The default value is `NoAuth`.

ProxyUserName User name/process that access the proxy.

ProxyUserDomain Domain of the proxy user.

ProxyPassword Password of the proxy user/process.

ProxyAccess Enables/disables the use of the proxy server. Disabled = `0`; Enabled = `1`. The default value is disabled.

Note: When using the daemon with proxy (i.e. ProxyAccess=1), it is recommended to set the MaxThreads for the HTTP server to 35 or below in order to avoid ephemeral port starvation.

2.1.3 [HTTP]

The [HTTP] section of the configuration file enables you to set whether ctIPd will interface with the querying device over HTTP protocol, as well as other options related to this interface type.

Enabled

The `Enabled` option indicates whether interfacing over HTTP protocol is enabled. The default value is `yes`.

Port

The `Port` option indicates the HTTP listening server port on which this communication will take place. The default value is: `8080`.

MaxThreads

The `MaxThreads` option defines the maximum size of the threads pool for processing new requests and for maintaining communication with awaiting HTTP connections. The default value is: `100`.

InitialThreads

The `InitialThreads` parameter defines the initial amount of threads that the system creates on startup. The system will create more threads, if needed, up to the number of maximum threads specified in the `MaxThreads` option. The default value is: `0`.

ReceiveTimeout

The `ReceiveTimeout` option specifies the amount of time in milliseconds that ctIPd will wait to receive data from the querying device over HTTP once a `classifyip` operation is initiated. The default value is `5,000`.

ListenBackLog

The `ListenBackLog` option defines the size of the TCP backlog queue. The default maximum value is `50`.

BindingAddress

The `BindingAddress` option defines the address to be bound to the socket. The default address is `INADDR_ANY`.

2.1.4 [UDP]

The [UDP] section of the configuration file enables you to set whether ctIPd will interface with the querying device over UDP protocol, as well as other options related to this interface type.

Enabled

The Enabled option indicates whether interfacing over UDP protocol is enabled. The default value is yes.

Port

The Port option indicates the port on which this communication will take place. The port can be a UNIX socket defined with an absolute path or a relative path to the current directory, or a numbered port. The default value is 5678.

MaxThreads

The MaxThreads option defines the maximum size of the threads pool for processing new requests. The default value is 100.

InitialThreads

The InitialThreads parameter defines the initial amount of threads that the system creates on startup. The system will create more threads, if needed, up to the number of maximum threads specified inthe MaxThreads option. The default value is 0.

ReceiveBuff

The ReceiveBuff option defines the maximum size, in bytes, of the UDP server socket receive buffer. The default value is 65,535.

MaxQueueSize

The MaxQueueSize defines the size of the queue for waiting requests. More requests will be discarded due to the connectionless nature of UDP. The default value is3,000.

RequestTimeout

The RequestTimeout defines the age threshold for discarding non-handled requests in the queue. The default value is 30 sec.

BindingAddress

The BindingAddress option defines the address to be bound to the socket. The default address is INADDR_ANY.

2.1.5 [RBL]

Enabled

The Enabled option indicates whether interfacing over RBL protocol is enabled. The default value is no.

DomainName

If RBL is enabled, the DomainName option should be defined to include the domain name of the ctIPd host.

Port

The Port option indicates the RBL listening server port on which communication will take place. The default value is: 53.

MaxThreads

The MaxThreads option defines the maximum size of the threads pool for processing new requests. The default value is 100.

InitialThreads

The InitialThreads parameter defines the initial amount of threads that the system creates on startup. The system will create more threads, if needed, up to the number of maximum threads specified inthe MaxThreads option. The default value is 0.

ReceiveBuff

The ReceiveBuff option defines the maximum size, in bytes, of the RBL server socket receive buffer. The default value is 65,535.

MaxQueueSize

The MaxQueueSize defines the size of the queue for waiting requests. The default value is 3,000.

RequestTimeout

The RequestTimeout defines the age threshold for discarding non-handled requests in the queue. The default value is 30 sec.

BindingAddress

The BindingAddress option defines the address to be bound to the socket. The default address is INADDR_ANY.

SOA

The SOA option defines the Start of Authority (SOA) resource record. The SOA resource record indicates that this DNS name server is the best source of information for the data within this DNS domain. Appropriate values for the following parameters can bedefined:

- DomainName

- `MailBoxDomainName`
- `Serial`
- `Refresh`
- `Retry`
- `Expire`
- `Minimum`

NS

The NS option enables you to define the name server. This is an optional field with no default value. Appropriate values are based on standard RFC 1035 guidelines.

TXT_TempfailFile

The `TXT_TempfailFile` option defines the name and location of a file containing the exact wording to use in the body of bounce-back messages when connection is tempfailed by the MTA as a result of a ctIPd recommendation. This option applies only to RBL/RBL+ transactions when TXT query is used.

The default location is the same directory as the ctIPd binaries, and the default name is: tempfail.txt. Therefore, the default value for the `TXT_TempfailFile` option is: ./tempfail.txt. Should you decide to move the files to a different directory, then the absolute path of that directory should be specified.

You can open the .txt file at any time and customize the contents. Within the file, you can use some predefined variables to trigger ctIPd to automatically include information such as the queried IP address or the transaction ID. For more information, see Customizing the TXT Response Contents.

TXT_PermfailFile

The `TXT_PermfailFile` option defines the name and location of a file containing the exact wording to use in the body of bounce-back messages when connection is permfailed by the MTA as a result of a ctIPd recommendation. This option applies only to RBL/RBL+ transactions when TXT query is used.

The default location is the same directory as the ctIPd binaries, and the default name is: permfail.txt. Therefore, the default value for the `TXT_PermfailFile` option is: ./permfail.txt. Should you decide to move the files to a different directory, then the absolute path of that directory should be specified.

You can open the .txt file at any time and customize the contents. Within the file, you can use some predefined variables to trigger ctIPd to automatically include information such as the queried IP address or transaction ID. For more information, see Customizing the TXT Response Contents.

2.1.6 [Stats]

The [stats] section of the configuration file determines whether or not statistics for the ctIPd's performance will be maintained. If enabled, statistics will be generated for a series of SNMP counters.

Enabled

The `Enabled` option enables you to control whether ctIPd maintains SNMP counters over HTTP, UDP and RBL (if enabled). The default value is `yes`.

Port

If the Enabled option is set to yes, the Port option will specify the server port. The default value is: /tmp/ctipd.stats. For Windows, use a numeric value only.

BindingAddress

The BindingAddress option defines the address to be bound to the socket. The default address is INADDR_ANY.

2.1.7 [DecisionManager]

Enabled

The Enabled option defines whether ctIPd will include a recommended call-to-action when responding to a query for IP reputation data. If enabled, ctIPd will weigh data from various sources and reply with one of three recommended actions: accept, permfail, or tempfail. The default option is yes.

# MaxRecords

The MaxRecords option defines the maximum number of IPs stored in the DM cache. Once the cache reaches the specified amount, the oldest records are overwritten by newer ones, thus limiting the cache records to the specified value set in the configuration file. The default value is: 100,000.

2.1.8 [LogServer]

LogMethod

The LogMethod option specifies which logging method is applied by ctIPd. There are three possible options:

Option	Description
0	When '0' is specified, no logging is performed.
1	When '1' is specified, the logging option is enabled and ctIPd locally maintains logs that enable you to generate reports. For more information, see the *ctIPD Reports* documentation. ***Note:*** *ctIPd uses an internal syntax for its reports.*
2	When '2' is specified, the logging data is included in the response to the querying device and passed through the ctIPd protocol. The logging data is not stored by ctIPd, locally. This enables you to import this data into an external logging and reporting system. For more information, see External Logging Options.

DirName

The DirName option specifies the directory name where the logfiles are located. By default, the files are stored in the /tmp/ctipd_log directory. If you decide to change this location to a directory that does not currently exist, ctIPd will search for the parent level directory and, if found, will create the directory as specified. If, however, ctIPd is unable to locate the parent directory you specified, ctIPd will store all logfiles in the default directory (/tmp/ctipd_log) and will issue an error.

DaemonName

The DaemonName option specifies the name of the daemon for which logfiles are maintained. The default value is `0001`.

Port

The `Port` option indicates the port via which logging data is transmitted. The default value is: `23456.`

MaxSize

The `MaxSize` option specifies the maximum combined storage size allocated for all logfiles. Once this limit is reached, older files are deleted as necessary to enable writing new files. If the `MaxPeriod` limitation is reached, the oldest files will be deleted even if the `MaxSize` value has not been reached. The default value is `2000` MB.

MaxPeriod

The `MaxPeriod` option specifies the maximum period of time to keep logfiles. Any logfile with an older creation date will be deleted from the storage. If the `MaxSize` limitation is reached, the oldest files will be deleted even if the `MaxPeriod` value has not been reached. The default value for this option is `24` hours.

MaxFileSize

The `MaxFileSize` option determines the maximum file size for any logfile. As soon as this value is reached, the file that is currently open for writing new records is closed and written to the storage, while a new logfile is created to begin logging data. The default value for the maximum logfile size is `50` MB.

Binding Address

The `BindingAddress` option defines the address to be bound to the socket. The default address is `INADDR_ANY`.

3 Running ctIPd

Once you unpack the .tar file and are ready to run ctIPd, you need to enter a command on the command line. Different switches exist to enable you to control the daemon. By default, ctIPd looks for its configuration file in `/etc/ctipd/ctipd.conf`. Therefore, if you wish to run ctIPd without modifying the file locations, you can use the following basic command line syntax to run ctIPd in interactive mode:

```
ctipd –i –c ctipd.conf
```

Alternatively, ctIPd can be run in the background. CYREN provides a sample script that can be placed in the appropriate init.d directory. The init.d script is a sample-only and may need to be adapted to your system.

3.1 Command Line Options

OPTIONS

-c <path>

Change the location of the configuration file by specifying –c and the new path.

-p <path>

Create a PID file and place it in the specified path location.

-i

Run ctIPd in interactive mode.

-s or –logfac

Determines which syslog facility to go to. Default value is the syslog facility number.

-v

View the ctIPd version number.

-h

Displays the help screen.

3.2 Stopping ctIPd

If you are running ctIPd interactively, you can kill the daemon using Ctrl-C. Alternatively, if you are not running it in interactive mode, you can send a SIGTERM to the process identified in ctipd.pid file or in the file created with the –p option above.

4 ctIPd Protocol

CYREN has developed a simple protocol to enable communication between the querying device and ctIPd. Using a pre-determined syntax for the envelope and data contained in a query, the querying device sends requests to ctIPd over HTTP, UDP, or RBL/RBL+. Once processed, the ctIPd responds to the request (or an error message, if the request could not be processed). The protocol structure for HTTP/UDP is extensible, meaning that the order of the headers is not mandatory and not all headers are required to be included in all responses.

The following sections detail the protocol options that can be used to initiate these requests by the querying device, and the response that will be sent by ctIPd with the most up-to-date information available on the sender IP.

Protocol and Syntax Options

ctIPd uses a simple protocol to control the transfer of data to and from a querying device. Communication between ctIPd and the querying device can be accomplished over:

- HTTP: standard HTTP/1.0 headers to POST requests are used in the following URL: http://<host>/ctipd/iprep.
- UDP: either a numbered port or UNIX socket over UDP is used.
- RBL/RBL+: the querying device passes and receives queries over DNS protocol.

When using HTTP or UDP, the same reputation data is sent to ctIPd as part of the request and the same data type values are returned in the response. When using RBL the zone contains the following data: queried IP, one or more data elements identifying the zone and the domain name of the ctIPd host machine.

A typical communication session over HTTP or UDP would begin with the querying device initiating a request to ctIPd asking for reputation data. The communication would include some or all of the following:

- The querying device sends a `classifyip` request for reputation data about a single IP address (hereafter, 'Sender IP' or 'Source IP'). The request contains envelope and data. The contents and syntax of the request is the same for interfacing over HTTP or UDP.
- ctIPd receives the request and responds to the `classifyip` request with a call-to-action (the default setting). Call-to-action values include: accept, permfail or tempfail). Alternatively, you can configure ctIPd to respond to the query with the call-to-action and specific values for several data types, or just the raw reputation data itself. For more information, see Appendix A: Raw Reputation Data. If a problem occurred and ctIPd is unable to process the query, it will respond with an error.
- Once the response to `classifyip` request is received, the querying device applies an appropriate flow control decision to the session, based on the information contained in the response. Some OEM partners use only the IP class to determine which decisions and/or actions their flow control application will apply to incoming traffic from a sender IP. Other OEM partners incorporate additional response data types in the configuration or policy options. The flow control application may also weight responses from multiple processes before applying its final decision.

To requests over the RBL/RBL+ interface, the response from ctIPd specifies whether the sender IP is found in a particular zone. Each zone stands for a different call-to-action. The following sections contain details on the protocol syntax used for all types of requests and responses, as well as explanations for the response data types contained in ctIPd's responses.

4.1 HTTP and UDP Requests and Responses

In the following sections, the request and response syntax are detailed. In the first section, a sample for all HTTP requests and responses is included. The second section provides information about each header.

4.1.1 Sample Requests and Responses over HTTP

classifyip Request over HTTP

URL: http://<host>/ctipd/iprep method: POST

```
POST /ctIPd/iprep HTTP/1.0
Accept-Language: en-us
Accept: */*
Content-Length: 3867
Host: 176.211.45.4
User-Agent: CYREN HTTP Client

x-ctch-request-id: 12345678
x-ctch-request-type: classifyip
x-ctch-pver: 1.0

x-ctch-ip: 216.163.176.213
```

Response to classifyip Request over HTTP

```
x-ctch-request-id: 81263
x-ctch-request-status: 0
x-ctch-pver: 1.0

x-ctch-iprange: 216.163.176.213-216.163.176.213
x-ctch-refid: 0001.0A090303.4693AF75.0114
x-dm-action:tmpfail
```

Response with Error over HTTP

```
HTTP/1.0 200 OK
Date: Sat, 12 May 2006 22:25:21 GMT
Server: CYREN IP Reputation daemon /1.0
```

```
Content-Length: 5421
Connection: close
Content-Type: text/plain

x-ctch-request-id: 12345678
x-ctch-pver: 1.0
x-ctch-request-status: 201

Your license key is invalid or blocked in the CYREN Datacenter.
Contact CYREN technical support.
```

4.1.2 Sample Requests and Responses over UDP

classifyip Request over UDP

```
x-ctch-request-id: 12345678
x-ctch-request-type: classifyip
x-ctch-pver: 1.0

x-ctch-ip: 216.163.176.213
```

Response to classifyip Request over UDP

```
x-ctch-iprange: 216.163.176.213-216.163.176.213
x-ctch-refid: 0001.0A090303.46D402B5.0016
x-ctch-dm-action:tmpfail
```

Response with Error

```
x-ctch-request-id: 12345678
x-ctch-pver: 1.0
x-ctch-request-status: 201

Your license key is invalid or blocked in the CYREN Datacenter.
Contact CYREN technical support.
```

4.2 classifyip Request for HTTP and UDP

Each classifyip request contains the request envelope as well as the request data. This enables you to define the IP address for each query.

Note: *All CYREN products support IPv6-format for handling IP addresses. ctIPd supports blocking of IPv6 addresses at the specific IP level and at defined IPV6 subnets. ctIPd can receive and process queries containing IP addresses in IPv6 format (as well as IPv4).*

Request Envelope

Header	Explanation
`x-ctch-request-id`	Optional header. This value is highly recommended in the case of UDP due to the connectionless nature of UDP.
`x-ctch-request-type`	Defines the request type.
`x-ctch-pver`	The current client version of the CYREN protocol.

Note: *The Request Envelope must be separated from the Request Data with an empty line.*

Request Data

Header	Explanation
`x-ctch-ip`	The IP address for which the query was sent. Each request can contain only one IP address.
`x-ctch-cache-only`	This optional header indicates if zero-latency is enabled. The default value is zero (0). If set to one (1), then zero latency response will be used.

4.3 Response to classifyip Request for HTTP and UDP

The response to classifyip request contains the IP reputation for IP range to which the queried IP belongs, as calculated by CYREN. Within the reply, various data types are used to determine the IP reputation ofthe calculated IP range and a reference ID is added to help track the transactions for diagnostics.

Response Envelope

Header	Explanation
`x-ctch-request-id`	The same as in the x-ctch-request-id headers used in the request
`x-ctch-request-status`	0 OK
	101 Unknown Command
	102 Bad Request
	201 Auth. Error
	301 Center Not Operational
	302 Connection to Datacenter failed
	303 Connection to Datacenter timed out
	304 Request to Datacenter timed out
	305 General Communication Error
	401 Internal Error
	501 Decision Manger Error
`x-ctch-pver`	The current server version of the CYREN protocol

Note: *The Response Envelope must be separated from the Request Data with an empty line.*

Response Data

By default, the response includes the IP range, the RefID and the call-to action. Alternatively, if configured to also forward raw definition date, additional information will be included in the response. For more information, see Appendix A: Raw Reputation Data.

Header	Explanation
`x-ctch-iprange`	Represents the first and last addresses of the IP range to which the queried IP belongs
`x-ctch-refid`	RefID of the transaction
`x-ctch-ipclass`	Calculated IP class of the Sender IP
x-ctch-volume	Calculated IP volume for the Sender IP
x-ctch-risk-level	Calculated risk level of the Sender IP
x-ctch-spam-ratio	Calculated Spam ratio for the Sender IP
x-ctch-valid-bulk-ratio	Calculated valid Bulk ratio for the Sender IP
x-ctch-maxhits	Maximum hits: Represents how long the reputation data for a specific IP is saved, based on the number of attempts to establish an SMTP connection. When implementing a local cache on the querying device, it is advised not to query ctIPd until maxhits is exceeded.
x-ctch-ttl	Time-To-Live (TTL): Represents how long the reputation data for a specific IP is saved, based on time. When implementing a local cache on the querying device, it is advised not to query ctIPd until the ttl is exceeded.
x-ctch-dm	Recommended call-to action. Options include: `accept`, `tempfail` and `permfail`.

Notes: *When a value for both x-ctch-maxhits and x-ctch-ttl are specified, it is recommended that the record be deleted from the local cache as soon as either value is reached.*

4.4 Response with Error for HTTP and UDP

The response with Error from ctIPd to the querying device contains the status and an explanation of why the error occurred. The same format is used to respond with errors for `classifyip` requests.

Response Envelope

Header	Explanation
`x-ctch-request-id`	Optional header
`x-ctch-pver`	The current server version of the CYREN protocol
`x-ctch-request-status`	Number representing a status. For example, for an invalid license key, you would receive: 201

Response Data

Header	Explanation
Free text	Full text with description of error. For example: Our license key is invalid or blocked in the CYREN Datacenter. Contact CYREN technical support.

4.5 Response Syntax

Each time ctIPd receives a `classifyip` request from the querying device over HTTP or UDP, it returns a response. By default, the response includes the following:

- IP range
- A call-to-action
- A RefID (the contents of which may vary depending on enabled actions within the configuration file).

Alternatively, you have the option to have ctIPd include raw data information in the body of the response. For more information, see: Appendix A: Raw Reputation Data.

Note: *ctIPd supports IP addresses in both IPv4 and IPv6 formats.*

4.6 Response Data Types

To offer the maximum flexibility to the OEM partner, CYREN answers each query with a detailed response containing several reputation-related data types. Included in this detailed response is the IP class, which represents a composite reputation value.

- Some OEM partners use only the IP class to determine which decisions and/or actions their flow control application will apply to incoming traffic from a sender IP.
- Others use only the Action contained in the ctIPd response without consulting additional reputation sources.
- Still other OEM partners choose to incorporate additional response data types in the configuration or policy options.

ctIPd's comprehensive response syntax offers maximum flexibility and ease of integration by enabling each OEM partner to make use of any part of the IP reputation response to determine an appropriate action regarding sender IP reputations.

4.6.1 Response Data Types

Each time ctIPd receives a `classifyip` request from the querying device over HTTP or UDP, it returns a response with values for each of the following datatypes:

- IP class
- IP range

- Call-to-Action
- RefID
- TTL

To requests over the RBL/RBL+ interface, the response from ctIPd specifies whether the sender IP is found in a particular zone.

4.6.2 IP Class Groups

The IP class data type represents a composite reputation value assigned by CYREN. There are three groups, each represented by a letter (R, T, or G) and a number. The number represents the level of risk.

- Groups R and T represent the volume received in a 24-hour period.
- Group G, the **Known Reputation** group, represents sources with fixed decisions, not based on their monitored volume. These include blacklisted, whitelisted and private IP sources.

A variety of elements are used to determine the risk value. By combining the volume and risk, CYREN has determined set IP classes that fall into three groups:

- High Volume
- Transitory or Low Volume
- Known Reputation

Group	Explanation
Rx: R1..R9	Represents sources with substantial volume history. For example, R1 being a source that was monitored over time with high volume and low risk and R9 a source with very high volume and high risk.
Tx: T1..T5	Represents sources with transitory volume or no volume history. For example, T1 being a source with low transitory volume and low risk and T5 a source with low transitory volume and high risk.
Gx: G1..G3	Represents sources for which there are fixed decisions and regardless of their monitored volume. In this group, G1 are all whitelisted sources, G2 are all blacklisted sources, and G3 are all private IP sources.

4.6.3 IP Range

While the querying device queries about a specific IP address, ctIPd will respond with a calculated range of IP addresses to which the queried IP was calculated to belong. The range is typically made of one or more successive IP addresses found to have the same behavior as the queried IP.

When the querying device generates a new request about a different IP address within the same range, then ctIPd will respond immediately using data from the local cache and will not forward the request to a CYREN datacenter unless the record in the local cache already expired for this IP range.

The IP range includes two sets of IP addresses:

- First IP in the range
- Last IP in the range
- Example: 1.2.3.0 – 1.2.3.255

Example of Private IP Addresses

From	To	Representation
10.0.0.0	10.255.255.255	10/8
172.16.0.0	172.31.255.255	172.16/12
192.168.0.0	192.168.255.255	192.168/16
127.0.0.0	127.0.0.255	127/8

4.6.4 Call-to-Action Options

If ctIPd's call-to-action capabilities are enabled (the default value), ctIPd analyzes available data, and returns one of the following call-to-action responses to the querying device:

Response	Explanation
accept	The IP address is not currently recognized as a “bad” source and therefore it is recommended that the pending SMTP connection be accepted.
tempfail	Enough data has been collected to suggest that the IP source may be a “bad” source but the data is not conclusive. It is recommended that the querying device will tempfail the connection allowing the source to reestablish the connection at a later time. All legitimate sources are equipped with a resending mechanism in response to tempfail response while only few bot applications are designed with such capabilities.
permfail	Enough data has been collected to determine that the IP source is responsible for sending spam or malicious content (phishing, virus, etc.) and therefore it is recommended that the SMTP connection be rejected.

4.6.5 RefID

The `RefID` is a reference ID record that will be used by CYREN to diagnose various support issues, per-transaction. It is recommended that you keep this RefID in case you need to trace back to determine why ctIPd returned any particular response. The syntax of the RefID changes according to the options that you have enabled within the ctIPd configuration file. At a minimum, it will contain the transaction number (`ctid`) and the call-to-action for the query. If enabled, it may also contain details of the raw reputation data and logging.

Note: *Without this key, CYREN is unable to retroactively determine the reason why a query was returned with any specific set of data values.*

4.7 RBL Queries and Responses

When a query is sent over RBL/RBL+ interface, ctIPd checks the specified zone within the request and, if a match is found in that zone, sends a response back, as detailed below.

RBL Zones

Client applications (querying devices) can send queries to ctIPd for one of several zones, per-query, using the standard RBL interface or for several zones in a single query using the RBL+ interface. When ctIPd receives a query, it looks at the zone name specified in the query and responds only regarding this zone.

Most devices implement standard RBL and for these cases the following zones are available:

- std.rbl.<domain_name> - keeps information about backlisted and 'grey' sources.
- black.rbl.<domain_name> - keeps information about blacklisted-only sources.
- grey.rbl.<domain_name> - keeps information about 'grey'-only sources

Note: *In this document, <domain_name> represents the specific domain name that is used for the machine that hosts ctIPd and it is different for every implementation. This value is determined in ctipd.conf within the [RBL] section.*

The RBL+ zone named plus.rbl.<domain_name> also keeps information about blacklisted and 'grey' sources and is used for responding with 127.0.0.2 for grey sources and 127.0.0.3 for black sources in a single response. This is different than the RBL zone, std.rbl.<domain_name> that responds with 127.0.0.2 for sources that are either blacklisted or 'grey' sources.

Typically, when querying an RBL server with a specific zone, the server checks its static lists on the specified zone and, if a match is found, responds with 127.0.0.2. However, in the case of ctIPd whereas the reputation of sources can change dynamically, the zone lists are not static. Additionally, for effectively blocking the 'bad' sources and allowing access to the 'good' sources, a sophisticated degree of throttling or rate limiting is required from ctIPd. This means that for the same source either the reputation changes over time or that the call-to-action changes although the reputation remains unchanged.

For example, for a source that attempts to connect multiple times, continuously, ctIPd may keep constant reputation as IPClass=T4 and RiskLevel=86 but the call-to-action will be different every few minutes as explained below:

- During the first 8 minutes after it was first queried, ctIPd will return 'tempfail' to all attempts to connect by this source.
- After the first 8 minutes, ctIPd will return 'accept' only to the first 5 messages each hour until there are no queries about this source for a period that is longer than 24 hours.

The following table summarizes all the information pertaining to the various RBL/RBL+ zones.

Zone	Explanation	Response	Action	IP Classes Correlation
std.rbl.<domain>	Queries for either blacklisted or 'grey' sources.	127.0.0.2	tempfail	G2, T3, T4, T5, R7, R8, R9
black.rbl.<domain>	Queries for blacklisted-only sources.	127.0.0.2	permfail	G2
grey.rbl.<domain>	Queries for 'grey'-only sources.	127.0.0.2	tempfail	T3, T4, T5, R7, R8, R9
plus.rbl.<domain>	Queries for both blacklisted and grey IPs and provides a different response, depending on the match found.	127.0.0.2	tempfail	T3, T4, T5, R7, R8, R9
		127.0.0.3	permfail	G2

The main advantage of the RBL/RBL+ interface is the ease of usage and deployment. However, the main disadvantage, by its design, is that when using this protocol for querying ctIPd, much of the reputation data is not returned in the response. For example, the following information is not included in RBL/RBL+ responses: IP Class, Risk Level, RefID for diagnostics, TTL for local caching on the querying device, and other statistical information such as the Volume, the Spam Ratio and the Valid Bulk Ratio.

These parameters along with the call-to-action are passed from ctIPd to the querying device when queried over HTTP/UDP interfaces. Another consideration for using the RBL/RBL+ interface is that the Decision Manager must be enabled (default) in order to use this protocol.

In the following sections, the query and response syntaxes for communication over RBL protocol are detailed. In the first section, a sample IP query over RBL and the corresponding response is included. The second section provides more detailed information about the query syntax options and the expected response.

4.7.1 RBL Sample

The querying device sends a standard DNS query to ctIPd containing the sender IP address and the requested RBL zone (see RBL Zones). ctIPd evaluates the query based on the RBL zone to which the query was directed and returns a response.

IP Query over RBL

```
dig @localhost -p 5353
   213.176.163.216.std.rbl.mycompany.com
```

Response to IP Query over RBL/RBL+

```
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27762
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0,
   ADDITIONAL: 0

;; QUESTION SECTION:
;213.176.163.216.std.rbl.mycompany.com.      IN    A

;; ANSWER SECTION:
213.176.163.216.std.rbl.mycompany.com. 0 IN   A
   127.0.0.2

;; Query time: 2 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Tue Jul 10 19:11:41 2007
;; MSG SIZE rcvd: 64
```

4.7.2 Response to IP Query over RBL/RBL+

The response over RBL/RBL+ contains Header, Question, and Answer sections.

[QUESTION SECTION]

The Question section repeats the query details, including the IP address and RBL/RBL+ zone information, and specifies the type of response it wishes to receive.

Note: *It is important to remember to formulate the query according to standard RBL/RBL+ syntax, such that the IP address is submitted in reverse order.*

Record Types

In the query, the querying device includes the record type. The Record Types field defines what type of response the querying device will receive, in cases where an IP connection should be rejected.

Record Type	Resulting Response if a Match is Found
A	Either 127.0.0.2 or 127.0.0.3 depending on the zone specified in the request.
TXT	Free text to explain to the sender why the SMTP session was terminated and the messages rejected. There are two options for the contents of the text. For more information, see Customizing the TXT Response Contents.
ANY	The ANY value contains both the IP address (A) and text (TXT). The IP address will depend on the zone specified in the request, and the TXT value is determined in the appropriate text file (for either tempfail or permfail). For more information, see Customizing the TXT Response Contents.

[ANSWER SECTION]

The Answer section of the IP response repeats the queried IP, details about the date, size and time of the query, the record and response result. When a query is sent over RBL/RBL+, ctIPd checks the specified zone and, if a match is found in that zone, sends a response back, as detailed above.

Customizing the TXT Response Contents

When ctIPd recommends either tempfail or permfail, a text message may be forwarded to the querying device to be relayed to the sender IP. This will happen only when sending a TXT DNS query type over RBL/RBL+ interface. The path and name of the two text files (one for each action), is defined in the configuration file and can be customized. The contents of the files can also be customized as needed.

The following table details the default message for each recommended action. Note below that variables are used within the text messages. These variables are dynamically UDPated to reflect the relevant information for each message.

File Name	Default Message Text
tempfail.txt	450 delivery from $IP is deferred and despite repeated attempts, this message could not be delivered. Try again later and if same response, then check your IP reputation at http://www.cyren.com/Site/Resources/Check_IP_Reputation.asp and make sure to provide the following reference code: $ctid
permfail.txt	550 delivery from $IP is rejected. Check your IP reputation at http://www.cyren.com/Site/Resources/Check_IP_Reputation.asp and make sure to provide the following reference code: $ctid

Variables in the Text Files

The following variables can be used anywhere in the text to present information that ctIPd will insert automatically before the message is sent to the IP sender.

Variable	Result
$IP	ctIPd will replace the variable with the queried IP.
$ctid	ctIPd will replace the variable with the unique transaction number assigned by ctIPd when the querying device sends the `classifyip` request.

Note: *If you change the text within these files, you must restart the daemon for your changes to be applied. ctIPd loads the configuration file and related values (such as these files) at the start of each session and will not reload any changed values to either the configuration file or these files until the daemon is restarted.*

4.8 External Logging Options

By default, ctIPd does not maintain logfiles detailing its operations. If you wish ctIPd to track this data, there are two options: either ctIPd can track this information using its own internal logging system and then generate reports (see the *ctIPD Logging and Reporting Guidelines* documentation), or ctIPd can export this data within the response to the querying device.

Note: *ctIPd's internal logging functionality is designed to enable ctIPd to create records and generate reports. While the syntax of the logs is for internal ctIPd use only, the reports themselves are explained in detail in the ctIPd Logging and Reporting Guidelines document.*

The logging data forwarded in the query is returned in a single data line separated by commas. Each element of data is preceded by a code to explain the meaning of the data followed by an equals sign (=) and then the associated value, as detailed in the following table:

Code	Explanation	Example
ts	Timestamp	2008-02-19 07:40:52
dn	The ctIPd daemon ID number	0001
ctid	The transaction number	1203406852.7F000001.2
tid	The CYREN Resolver transaction number	0001.0A0B0302.47BA89D4.0019
fc	Flag indicating whether the recommendation was taken from the local cache, as follows: 0=No; 1=Yes	1
ip	IP address being queried	1.1.2.3

Code	Explanation	Example
dm	Flag indicating whether the decision manager options are enabled and what the recommended action is, as follows: 0=disabled 1x=enabled and accept 2x=enabled and tempfail 3x=enabled and permfail	201
fs	Indicates when this IP was first queried. Time represented in seconds from Epoch	1203406852
ipc	IP Class	T2
rsk	Risk Level	66
pt	pt = <protocol type>: 0=HTTP 1=UDP	0
zl	Zero latency enabled, as follows: 0=No; 1=Yes.	0
cl	cl = <child license key if authenticated mode: license[:oem token]>	0

To implement an external logging system:

1. In the [LogServer] section of the ctipd.conf file, modify the value of `LogMethod` to 2.
2. Restart the daemon.

Note: *If you modify the ctipd.conf file, you must restart the daemon. The logging process will only begin after you restart the daemon.*

5 SNMP Counters

ctIPd maintains a series of SNMP counters to monitor the daemon's performance and to log its activities. The counters are divided into the following groups: HTTP, UDP, RBL/RBL+, Decision Manager, and IPREP counters. The counters are described in the `./ctipd/snmp/mibs` directory and enabled/disabled in the configuration file. To use the SNMP counters you can configure a special port in the configuration file, as described in the [stats] section. Open the port to the stats server, send the string `getall`, and ctIPd will return all available data for all counters. A sample of such sub-agent is supplied by CYREN in the package.

5.1 General IP Reputation Counters

Following is a list of counters related to IP reputation services over both HTTP and UDP:

SNMP Counter	Explanation
uptime	Total number of seconds elapsed from the time the daemon was started.
pid	The daemon's process ID.
totalCenterRequests	The total number of queries forwarded to the Datacenter.
totalCommErrors	The total number of communication errors that occurred while trying to query the Datacenter.
totalCenterRequestTime	The total amount of time the client waited for a response from the Datacenter.
ipRepCacheSize	The number of records in the local cache.
ipRepCacheTotalHits	The total number of a response to a query was found in the local cache.
ipRepCacheTotalMisses	The total number of times a query response was not found in the local cache.
ipRepCacheRecords	Total number of records in the cache (both HTTP and UDP).
ipRepTotalClassifyIpCurrRequests	Total number of ClassifyIP queries currently being processed.
ipRepTotalClassifyIpCenterRequests	Total number of `classifyip` queries (both HTTP and UDP) forwarded to the Datacenter.
ipRepTotalClassifyIpErrors	The total number of general errors in ClassifyIP queries.

5.2 HTTP Counters

Following is a list of SNMP counters related to communication over HTTP:

SNMP Counter	Explanation
ipRepHttpCurrRequests	Number of HTTP requests that ctIPd is currently processing.
ipRepHttpQueueSize	Number of HTTP connections waiting to be processed.
ipRepHttpTotalWaitTime	Total wait time to date of all queries in the queue.
ipRepHttpTotalClassifyIpRequest	Total number of queries to date that have been sent to ctIPd.
ipRepHttpTotalClassifyIpErrors	The total number of errors that resulted from ClassifyIP queries to ctIPd.
ipRepHttpTotalClassifyIpTime	The total amount of processing time for ClassifyIP queries.

5.3 UDP Counters

Following is a list of SNMP counters related to communication over UDP:

SNMP Counter	Explanation
ipRepudpCurrRequests	The number of UDP queries that ctIPd is currently processing.
ipR ipRepepudpQueueSize	The total size of the queries queue.
ipRepudpTotalWaitTime	Total wait time for all queries in the queue.
ipRepudpTotalClassifyIpRequests	The total number of ClassifyIP queries.
ipRepudpTotalClassifyIpErrors	The total number of errors that occurred as a result of ClassifyIP queries to ctIPd.
ipRepudpTotalClassifyIpTime	Total amount of processing time on `classifyip` queries.

5.4 RBL Counters

Following is a list of counters related to ctIPd performance over RBL/RBL+:

SNMP Counter	Explanation
ipRepRblCurrRequest	The number of RBL queries currently being processed.
ipRepRblQueueSize	The size of the RBL connections queue.
ipRepRblTotalWaitTime	Total wait time for all RBL queries in the queue.
ipRepRblTotalClassifyIpRequest	The total number of ClassifyIP queries sent via RBL.

SNMP Counter	Explanation
ipRepRblTotalClassifyIpErrors	The total number of errors that occurred as a result of ClassifyIP queries.
ipRepRblTotalClassifyIpTime	The total amount of processing time for ClassifyIP queries.

5.5 Action Counters

Following is a list of action counters related to the recommendations that ctIPd returns to the querying device:

SNMP Counter	Explanation
ipRepDecisionManagerTotalRequests	Total number of queries (over all protocols) that have been processed by ctIPd.
ipRepDecisionManagerTotalTempFail	Total number of tempfail responses.
ipRepDecisionManagerTotalAccept	Total number of accept responses.
ipRepDecisionManagerTotalPermFail	Total number of permfail responses.
ipRepDecisionManagerCacheSize	Total number of records currently in the local Decision Manager cache.

6 Appendix A: Raw Reputation Data

By default, ctIPd returns a call-to-action, recommending that the querying device take one of three actions: accept, tempfail, or permfail. For OEM partners wishing to disable the built-in Decision Manager functionality and instead have their application compute the call-to-action, CYREN has an option to include a full range of raw reputation data within the response to `classifyip` requests.

Should the OEM partner wish to receive the raw reputation data, this option must be enabled in the configuration file.

This section details how to enable responses with raw reputation data, as well as a detailed explanation of each of the data types included in the response.

6.1 Sending Raw Reputation Data

Once enabled, every response to `classifyIP` requests over HTTP/UDP interfaces will include the raw reputation data.

Note: *If the Decision Manager is enabled, ctIPd's calls-to-action are also included in the response.*

If the logging method is set to '2', then each response's log information will also be included in the response.

To enable ctIPd to send raw data files:

1. In the [IPRep] section of the ctipd.conf file, change the value of `SendRawData` to `yes.` The default value is `no.`
2. Restart the daemon.

Note: *For more information on the configuration file and options related to configuring the logging functionality, see ctIPd Configuration, as well as the ctIPd Logging and Reporting Guidelines document.*

6.1.1 Raw Reputation Data Types

ctIPd enables you to specify that you would like to have the raw reputation data included in the response send to the querying device.

Note: *This option is not applicable to RBL/RBL+ protocol.*

The following raw reputation data types are included in the raw reputation data:

- IP class
- IP volume
- Risk level
- IP Spam ratio
- IP valid Bulk ratio
- TTL

6.1.2 IP Class

The IP class data type represents a composite reputation value assigned by CYREN. There are three groups of IP classes, each represented by a letter (R, T, or G) and a number. The number represents the level of risk.

- Groups R and T represent the volume received in a 24-hour period.
- Group G, the Known Reputation group, represents sources with fixed decisions, not based on their monitored volume. These include blacklisted, whitelisted and private IP sources.

A variety of elements are used to determine the risk value. By combining the volume and risk, CYREN has determined set IP classes that fall into three groups:

- High Volume
- Transitory or Low Volume
- Known Reputation

Group	Explanation
Rx: R1...R9	Represents sources with substantial volume history. For example, R1 being a source that was monitored over time with high volume and low risk and R9 a source with very high volume and high risk.
Tx: T1...T5	Represents sources with transitory volume or no volume history. For example, T1 being a source with low transitory volume and low risk and T5 a source with low transitory volume and high risk.
Gx: G1...G3	Represents sources for which there are fixed decisions and regardless of their monitored volume. In this group, G1 are all whitelisted sources, G2 are all blacklisted sources, and G3 are all private IP sources.

For information on the logic ctIPd's internal decision manager uses to determine the recommended call-to-action, see the *ciIPd Implementation Guidelines document*.

6.1.3 Volume

The IP volume is a 3-value array as detailed below:

- Recent Peak, indicates the deviation of the recent volume from the approximated hourly peak calculated from the daily average over the last 30 days [signed integer].
- Calculated daily average of the last 30 days, ranging from ‘0’ to ‘10’ with ‘0’ being a source IP having no monitored traffic [unsigned integer].
- Standard deviation (stddev) of the last 24 hours from the daily average of last 30 days, ranging from ‘0’ to ‘10’. [unsigned integer].

6.1.4 Risk Level

This value represents the risk level, a composite value that weights a host of variables on the Datacenter-side. A value of ‘0’ in the Risk Level represents a whitelisted IP and a value ‘100’ represents a blacklisted IP.

6.1.5 Spam Ratio

The Spam ratio is calculated by dividing the monitored spam volume by the total monitored volume. The Spam ratio is a 2-value array as detailed below:

- Recent Peak, indicates the deviation of the recent Spam ratio from the approximated hourly peak calculated from the daily average over the last 30 days, ranging from ‘0’ to ‘100’ [percentage].
- Calculated daily average of the last 30 days, ranging from 0 to 100 with ‘0’ being a source IP having no monitored Spam traffic [percentage].

6.1.6 Valid Bulk Ratio

The valid Bulk ratio is calculated by dividing the monitored valid Bulk volume by the total monitored volume. The valid Bulk ratio has a single value as detailed below:

Calculated daily average of the last 30 days, ranging from ‘0’ to ‘100’ with ‘0’ being a source IP having no monitored valid Bulk traffic [percentage].

6.1.7 Time To Live (TTL)

The TTL value determines how long ctIPd will reuse reputation data before sending a new query for the same source IP. The TTL can be determined by ctIPd either as a time unit (e.g., 2 hrs.) or max hits (e.g., 8 times).

6.2 Response to classifyip with Raw Reputation Data

The response to classifyip request contains the IP reputation for IP range to which the queried IP belongs, as calculated by CYREN. Within the reply, various data types are used to determine the IP reputation of the calculated IP range and a reference ID is added to help track the transactions for diagnostics.

Response to classifyip Request over HTTP with Raw Reputation Data

```
x-ctch-request-id: 81263
x-ctch-request-status: 0
x-ctch-pver: 1.0
```

```
x-ctch-iprange: 216.163.176.213-216.163.176.213
x-ctch-refid: 0001.0A090303.4693AF75.0114
x-ctch-ipclass: T3
x-ctch-volume: 0,3,3
x-ctch-risk-level: 73
x-ctch-spam-ratio: 100,42
x-ctch-valid-bulk-ratio: 57
x-ctch-maxhits: 3
x-ctch-ttl: 7126
x-dm-action:tmpfail
```

Response to classifyip Request over UDP with Raw Reputation Data

```
x-ctch-iprange: 216.163.176.213-216.163.176.213
x-ctch-refid: 0001.0A090303.46D402B5.0016
x-ctch-ipclass: R7
x-ctch-volume: 0,10,10
x-ctch-risk-level: 78
x-ctch-spam-ratio: 87,87
x-ctch-valid-bulk-ratio: 0
x-ctch-24h: 468452,53
x-ctch-ttl: 1800
x-ctch-from-cache: no
x-ctch-dm-action:tmpfail
```

Response Envelope

Header	Explanation
`x-ctch-request-id`	The same as in the x-ctch-request-id headers used in the request
`x-ctch-request-status`	0 OK
	101 Unknown Command
	102 Bad Request
	201 Auth. Error
	301 Center Not Operational
	302 Connection to Datacenter failed
	303 Connection to Datacenter timed out

Header	Explanation
	304 Request to Datacenter timed out
	305 General Communication Error
	401 Internal Error
	501 Decision Manager Error
`x-ctch-pver`	Current server version of the CYREN protocol

Note: *The Response Envelope must be separated from the Request Data with an empty line.*

Response Data

By default, the response only includes the IP range, the RefID and the call-to action. Alternatively, if configured to also forward raw definition date, additional information will be included in the response.

Header	Explanation
`x-ctch-iprange`	Represents the first and last addresses of the IP range to which the queried IP belongs
`x-ctch-refid`	RefID of the transaction
`x-ctch-ipclass`	Calculated IP class of the Sender IP
`x-ctch-volume`	Calculated IP volume for the Sender IP
`x-ctch-risk-level`	Calculated risk level of the Sender IP
`x-ctch-spam-ratio`	Calculated Spam ratio for the Sender IP
`x-ctch-valid-bulk-ratio`	Calculated valid Bulk ratio for the Sender IP
`x-ctch-maxhits`	Maximum hits: Represents how long the reputation data for a specific IP is saved, based on the number of attempts to establish an SMTP connection. When implementing a local cache on the querying device, it is advised not to query ctIPd until maxhits is exceeded.

Header	Explanation
`x-ctch-ttl`	Time-To-Live (TTL): Represents how long the reputation data for a specific IP is saved, based on time. When implementing a local cache on the querying device, it is advised not to query ctIPd until the TTL is exceeded.
`x-ctch-dm`	Recommended call-to action. Options include: accept, tempfail and permfail.

Notes: *When a value for both x-ctch-maxhits and x-ctch-ttl are specified, it is recommended that the record be deleted from the local cache as soon as either value is reached.*