# The Velocite Systems Protected Point to Point Solution

## *Functional Specification*

Updated: Oct 27, 2011

# Table of Contents

# 1 Executive Description

Computer network systems are being used for growing quantities of critical organization data. The transport of this data is traditionally protected by a private link, or a Virtual Private Network (VPN) over the Internet. The remainder of this document shall use the VPN term as representative of all such private links. General purpose VPNs are capable of interfacing with multiple clients, but at a cost. For example, when used between data centers, the performance suffers.

For long lasting connections between data centers, the encryption keys need to be replaced periodically to ensure that the VPN is secure. The traditional VPN method for doing this is to close the existing VPN session and renegotiate the key exchange. Because of the performance cost, this is not done frequently. Also, the act of renegotiating the keys signals to any observer that the keys are being replaced.

The Velocite Systems Protected Point to Point ($P^3$) Solution creates an encrypted, tunnel connection between data end points (data centers, server–to–server, server to client, …) that replaces keys **WITHOUT** closing the encrypted session. The control information to accomplish this is carried within the encrypted session, making it impossible to determine that is occurring from simple monitoring of the connection.

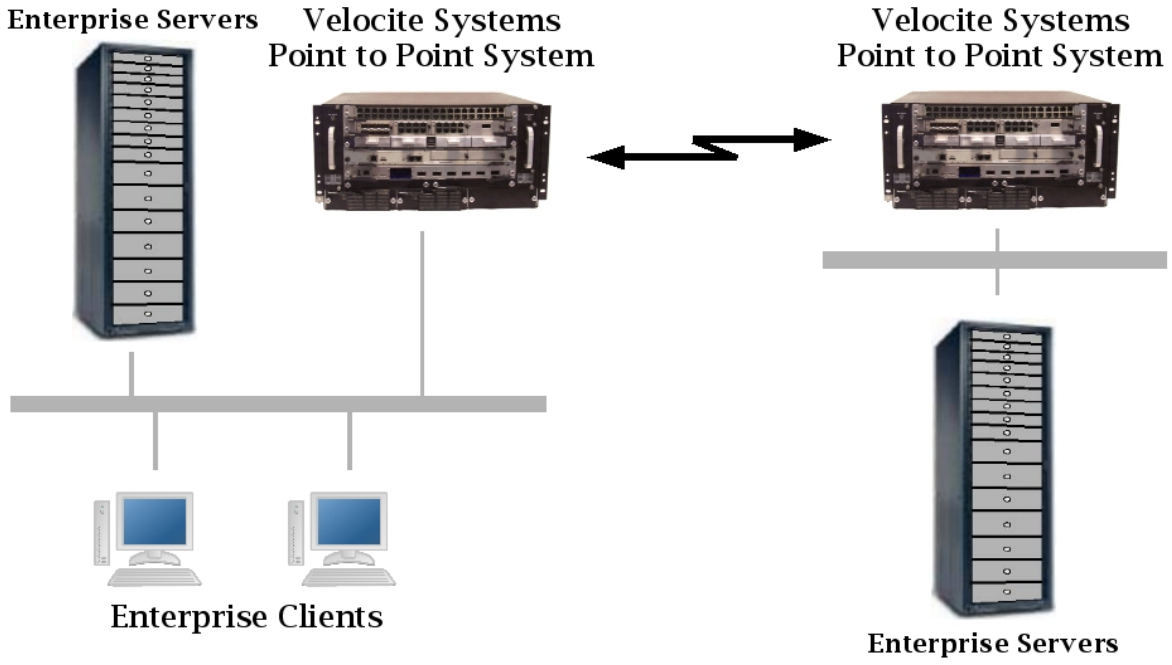


**Figure 1: Velocite Systems Protected Point to Point Solution**

Velocite Systems $P^3$ Systems operate in pairs to establish an encrypted Point–to–Point connection. The encrypted session uses either AES 128 or AES 256 encryption keys, and these keys are dynamically replaced frequently to make it extremely difficult to break the encryption of the link.

The Velocite Systems Protected Point to Point ($P^3$) Solution further increases the safety of replacing keys by manipulating the data prior to being encrypted. This makes it difficult to target fields within the data or even to recognize whether the data has been correctly decrypted, which raises the cost of attempting to break the encryption keys.

Velocite Systems $P^3$ Systems may be configured as a cluster group to share a single IP address. This provides load balancing and failover.

# 2 Protected Point to Point Solution

Velocite Systems P³ Systems operate in pairs to establish an encrypted Point–to–Point connection. There is a primary and secondary in each pairing. The encrypted session uses either AES 128 or AES 256 encryption keys.

The primary P³ Solution produces new keys that are delivered to the secondary P³ System to be used for encrypting network sessions. These keys are changed at regular intervals, to provide a secure channel without disrupting the session. To ensure their protection, an encrypted control channel within the session is established to handle the dynamic replacement of keys, to make it extremely difficult to break the encryption of the link.

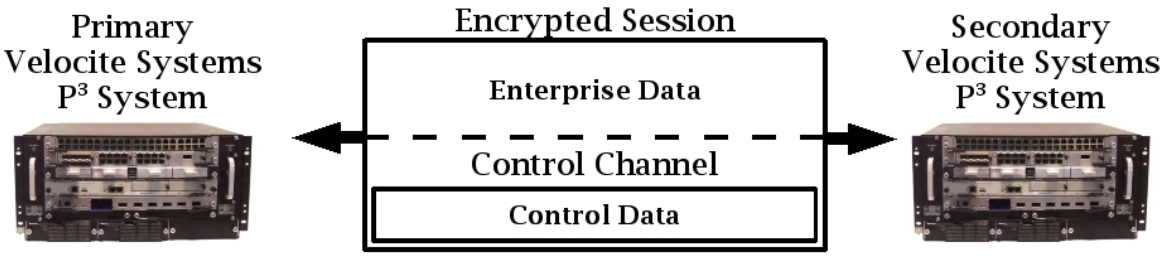


**Figure 2: Secure Data and Control Communications**

## 2.1 Application

The Velocite Systems P³ Solution performs six functions. Of these, only the Administration interface is seen by administrators, the rest occur within the P³ Systems. The functions are:

1) Administration
2) Initialization
3) Packet Intercept and Routing
4) Session processing

5) Encryption ReKeying

6) Session recovery

## 2.1.1 Administration

The Velocite Systems Point to Point Protected Solution administration application automates the management of $P^3$ primary and secondary systems as much as possible. This provides for dynamic and permanent updates to $P^3$ system definitions as well as status reports and session management..

Primary $P^3$ systems may connect to thousands of secondaries, each of which requires a unique configuration. To make managing this possible, administration of multiple secondary systems simultaneously is provided. Individual secondaries may be members of multiple groups which are defined to have common configuration characteristics. There is still flexibility for unique configuration parameters of $P^3$ systems within a group.

The initial deployment of a secondary system requires the IP address of the primary $P^3$ System for the secondary to establish a connection. This is set when the secondary $P^3$ System application is installed, but may be modified later.

There are three functions handled by the $P^3$ System Administration:

1) **Administration:** This function includes user administration, status reports, and system configuration and is described in the System Administration section.
2) **Configuration:** Both the primary and secondary configuration requires local and remote host definitions. The primary configuration includes key management definitions while the secondary does not. These definitions are described in the Primary and Secondary Configuration sections.
3) **Session Handling:** This function includes requesting status about connections, forced rekeying, and session shutdown. This is described in the Session Management section.

The user interface component of the $P^3$ System runs as a standalone application. Updates to the $P^3$ System application are implemented using interprocess communication. The Core Library Application Programming Interface section defines the specific commands used for this communication.

### 2.1.1.1 Session Management

The P3 user interface allows dynamic management of the primary or secondary. This includes managing sessions, updating configuration data, getting status, and performing operations on keys.

1) **Anonymous connections:** Secondaries may make anonymous connections to the primary to provide configuration data. The procedure for accomplishing this is as follows:
2) **Status:** Status may be obtained from the primary on individual sessions or all secondaries belonging to a group for the following:
3) **Keys:** An administrator may perform the following on keys:
    1) Generate a new key array
    2) Force rekeying for selected sessions
    3) Flush a key array and optionally generate a new one
4) **Session shutdown:** Individual sessions or all secondaries belonging to a group may have their sessions shutdown.

### 2.1.2 Initialization

The Velocite Systems $P^3$ Solution initialization conforms to the IPsec standard defined in RFC 4301 and its supporting standards. The protocol is modified to enable the encryption keys to be replaced dynamically.

A secondary $P^3$ System makes the connection request for the main session. Secondaries will not accept connection requests themselves.

When the main session is established, the primary $P^3$ System initializes the Control Channel using messages defined for the Velocite Systems $P^3$ Solution. The messages transmitted through the Control Channel appear to be normal data, but are encrypted using keys that are different from the data session.

### 2.1.3 Packet Intercept and Routing

Each $P^3$ System must intercept packets with the IP and other protocol headers intact. The packets are then encrypted and forwarded to the correct $P^3$ System, which decrypts the packet and forwards it to the local network.

If a primary $P^3$ System establishes sessions with more than one other $P^3$ System, it routes packets to the correct $P^3$ System through the $P^3$ session. To accomplish this it uses a $P^3$ System routing table to match the destination IP address of each packet to the correct secondary $P^3$ System.

In the outbound packet intercept process, the primary $P^3$ System intercepts each packet destined for another $P^3$ System with the IP and other protocol headers intact. If there are more than one connections established, the destination address is matched using the $P^3$ System routing table definition. Then the full packet is encrypted and forwarded to the correct $P^3$ System.

In the inbound packet intercept process, the encrypted packet data is decrypted. If the destination IP address of the packet is the P³ System, the packet data is handled specially. Otherwise, the full packet is forwarded to the local network.

The P³ Systems special handling of control channel data includes encryption and decryption using the control channel encryption key and bypassing the TCP/IP stack to manage the unencrypted data.

In the outbound control packet process, predefined IP and TCP headers are prepended to the control data. Then the full packet is encrypted and the result is treated like normal data.

In the inbound control packet process, the full packet is decrypted. The predefined IP and TCP headers are removed from the control data and the data is forwarded directly to the control application to be processed.

### 2.1.4 Session Processing

The Velocite Systems P³ Solution supports only one usage scenario defined in the IKEv2 specification, which is the Security Gateway to Security Gateway Tunnel.

The primary P³ System may have sessions with multiple secondary P³ Systems. If so, packets received from the local network are routed to the correct secondary P³ System, as described in the Packet Intercept and Routing section.

The full IP packets received from the local network are encrypted and used as the payload data of the encrypted packet. These packets are structured using a special P³ header. The Velocite Systems P³ Solution requires the addition of a 32–bit field at the end of the P³ header and immediately before the encrypted data. The format of IP version 4 encrypted packets is:

```
| 20 octets | 4 octets | 4 octets | variable length |

| IP Header | P3 Flags | P3 ID    | Encrypted Data  |
```

The format of IP version 6 encrypted packets is:

```
| 40 octets | 4 octets | 4 octets | variable length |

| IP Header | P3 Flags | P3 ID    | Encrypted Data  |
```

where the fields are:

1) IP Header: The standard version 4 or version 6 IP header with the protocol field set to 50 (ESP).
2) P³ Flags: Currently reserved.

3) $P^3$ ID: The Identification number of the packet is an unsigned 32–bit integer. It indicates which encryption key is used to decrypt the packet payload.

Data received by one $P^3$ System from the other is decrypted, which may be handled in hardware. The data is then either forwarded to its destination or the Control Channel Decryption process.

When Control Channel data is to be sent to another $P^3$ System, predefined TCP and IP headers are prepended. Only the address fields are processed by the receiving $P^3$ System, so the headers may be statically defined. The entire packet is encrypted using the control encryption key, and it is then handled the same as a data packet.

If, after decrypting a packet, the destination is seen to be the $P^3$ System itself, the packet is decrypted using the control encryption key. Then the IP and TCP headers are removed and the data is forwarded to the Control Channel application.

## 2.1.5 Data Encryption

The Velocite Systems $P^3$ Solution increases the safety of transporting keys over the same session as data in three ways:

1) **Limit differences in packet sizes**
2) **Add pad characters from existing data**
3) **Relocation of data within a packet**

## 2.1.6 Encryption ReKeying

The Velocite Systems Encryption ReKeying Process replaces encryption keys frequently. When this is to occur, the primary $P^3$ System notifies the secondary, supplies the new keys, and indicates the ReKeying point.

The primary $P^3$ System generates new keys for both the main session and the Control Channel. The keys are replaced using Velocite Systems $P^3$ Solution control messages.

The primary $P^3$ System sends a control message with the data and control key values. When the secondary receives the message, it updates the keys it is using and returns a status message. If the status is good, the primary updates the keys it is using and sends a message to indicate the first packet it will encrypt using the new keys. The secondary returns a message indicating the first packet it will encrypt using the new keys.

If the status is not success, the primary attempts again for up to four times. If this does not result in success, Session Recovery is performed.

## 2.1.7 Session recovery

There are two Velocite Systems $P^3$ Systems configuration modes that may be used together or individually to reduce the likelihood of failure:

1) **Clustering:** In this mode, there are multiple $P^3$ Systems that share the same IP address.

2) **Failover:** In this mode, a backup takes over the functions of the main $P^3$ System.

If a session is lost in spite of the backup configuration, the primary $P^3$ Solution performs the following steps:

1) **Login termination:** When a session failure occurs, all enterprise connections to the primary $P^3$ System are immediately terminated. In the case of TCP sessions, this should be done by sending a Reset. For other protocols, an ICMP Destination Unreachable packet may be generated.

2) **Notify administrators:** An error message with all available information on the failure is logged to the enterprise logging facility to notify administrators of the failure.

If a session is lost in spite of the backup configuration, the secondary $P^3$ Solution performs the following steps:

1) **Login termination:** When a session failure occurs, all enterprise connections to the secondary $P^3$ System are immediately terminated. In the case of TCP sessions, this should be done by sending a Reset. For other protocols, an ICMP Destination Unreachable packet may be generated.

2) **Session retry:** After a brief wait, the $P^3$ System attempts to automatically re–establish the session with the primary $P^3$ System and continues to do this periodically.

    1) The first time the $P^3$ System sets the wait period to 5 seconds, and after that period, attempts to connect to the primary.

    2) Each time the connection cannot be established, 5 seconds is added to the wait time and after that period, the connection is attempted.

    3) When the wait period is 1 minute, the wait time is not increased, so that the secondary will attempt to connect to the primary once a minute.

# 3 Installation

The Velocite Systems $P^3$ Solution is essentially a turnkey system. However, some installation configuration is required. All $P^3$ Systems must have a routing table defined to be able to forward packets to or from the local network correctly. This is accomplished with a script that prompts the administrator for IP addresses.

Each secondary $P^3$ System must have the IP address of the primary $P^3$ System it is to request a connection with defined. Also, the initial encryption configuration must be set. This is included in the script.

Primary $P^3$ Systems have a more complex configuration, and an application to manage this is included. Administrators can run this application for initial configuration as well as maintenance and troubleshooting.

# 4 Performance

## 4.1 Throughput

The throughput of the Velocite Systems P³ Solution is equal to the throughput of traditional network encryption over equivalent connection infrastructure. The minimum period of key generation to ensure the security of the sessions is once every ten minutes for every session.

## 4.2 Scalability

The Velocite Systems P³ Solution system includes two options to allow the required scalability for an enterprise:

1) **Optional Hardware Key Generation:** If the enterprise environment requires a large number of encryption keys to be generated frequently, the Key Servers may be fitted with Hardware Key Generation.

2) **P³ Solution Load Balancing mode:** In this mode, all P³ Solutions are active and share an IP address. They also mirror each other's storage or are connected to a Storage Area Network. There is one primary P³ Solution that manages connection requests and forwards the request to the next available P³ Solution.

## 4.3 Availability

The P³ Solutions supports the following failover mode of operation:

1) **P³ Solution Failover mode:** In this mode, a primary and one or more backup P³ Solutions are operational. If the primary P³ Solution fails, the designated backup automatically becomes the primary. Secondary P³ P3 Solutions re–establish network sessions with the new primary. When the original primary recovers, it becomes the designated backup, and may become the primary by using an administrative command.

### 4.3.1 Clustering

The purpose of creating a cluster group is to improve performance on a high volume network. In this mode, multiple P³ Systems share the same IP address and the processing load is distributed.

To implement a cluster group, the operating system must support the feature. (For Linux, see http://www.austintek.com/LVS/LVS–HOWTO/HOWTO.)

The structure of the clustering configuration includes:

1) **Director:** The Director accepts packets and routes them to the appropriate group member. This may be handled by one of the $P^3$ Systems or it may be a separate unit. The former should include a failover capability.
2) **Realserver:** The Realserver is the group member that processes packets.

A common use of clustering is for web server farms. In this application, many sessions are established, and the Director sends all packets of a single session to the same Realserver. The Velocite Systems secondary $P^3$ System establishes a single session with the primary and all packets of this session may be distributed to any of the Realservers. Therefore, the Director algorithm for distributing to Realservers may need to be modified.

Because the application does not actually process any data except that of the Control Channel, there does not need to be an awareness of the application state between members of the cluster group. To share control information, the Velocite Systems application must establish a connection between all members of the group. The method for sharing this information is different for primary and secondary $P^3$ Systems.

# 5 Troubleshooting

## 5.1 Messages

All failure messages issued by either the primary or secondary $P^3$ Solution include a reason code to assist administrators in troubleshooting or reporting the problem.

## 5.2 Support

Velocite Systems Standard, Extended, and Premier support options are available.

## 5.3 Software Updates

The Velocite Systems $P^3$ Solution provides a single update manager for all of the platform's software components. The applications installed by the users of the platform are not automatically included in that package management.

# 6 Standards Compliance

The Velocite Systems $P^3$ System conforms to the following security standards:

1) NIST FIPS 140–2 standard for encryption

The implementation of the Velocite Systems $P^3$ System conforms to the following Internet Engineering Task Force (IETF) standards:

**1) IP Security**

1) RFC 4301: Security Architecture for the Internet Protocol
2) RFC 3723: Securing Block Storage Protocols over IP

**2) Cryptographic Algorithms**

1) Internet Key Exchange (IKE)
2) RFC 3686: Using AES Counter Mode With IPsec ESP
3) RFC 4309: Using AES CCM Mode With IPsec ESP
4) RFC 4434: AES–XCBC–PRF–128 algorithm for IKE
5) RFC 4615: AES–CMAC–PRF–128 Algorithm for IKE

# 7 Glossary

The following terms and abbreviations are used in this document.

**Protected Point to Point ($P^3$) Solution:** The hardware and software configured to provide a secure connection over a public, routed network. The security is provided by establishing an encrypted session between two $P^3$ Systems and encapsulating all packets forwarded to them in encrypted packets, making it appear that all traffic between the two is a single, encrypted session.

**Protected Point to Point ($P^3$) System:** The hardware device used as an endpoint in a $P^3$ session. This includes all necessary network interfaces and specialized hardware for performing tasks, such as encryption.

**Control Channel:** An encrypted session established between the two $P^3$ Systems used to exchange control information between them. The session packets are treated the same as any other data from the perspective of the $P^3$ Session. This session is encrypted with keys that are different from those used for the $P^3$ Session.

**Velocite Systems Encryption ReKeying Process:** The process of replacing encryption keys for both the Encrypted Session and the Control Session without closing and restarting either of those sessions. The information for the new keys is transmitted using the Control Session.

**IP Security Architecture (IPsec):** A protocol suite for securing Internet Protocol (IP) communications by authenticating and encrypting each IP packet of a data stream. IPsec also includes protocols for establishing mutual authentication between agents at the beginning of the session and negotiation of cryptographic keys to be used during the session.

**Advanced Encryption Standard (AES):** The encryption standard adopted by the US government. It includes three key ciphers: AES–128, AES–192, AES–256. Each cipher has a 128 block size, with key sizes of 128, 192, and 256, respectively.

**Clustering:** A technique of grouping computers to act as if they were a single computer. In particular, a single network address is shared between them, and packets destined for that address may be processed by any of the computers in the cluster group.

**Failover:** A technique for providing the capability to automatically start using a redundant, backup computer in the event of the failure of the currently active computer. This requires most or all applications to continue processing without interruption.

# Appendix A: Configuration

There is a single Velocite Systems P$^3$ Solution configuration file for either a P$^3$ Primary or Secondary. Both have two sections, the first defining the local host and the second defining the remote host(s).

The syntax of the configuration files is:

1) Lines with "#" at the beginning are comments
2) Blank lines are ignored
3) Every other line defines a keyword, and the entire definition must be on the same line

## 1 Primary

The local section of the primary configuration must begin with the keyword "localstart" and end with the keyword "localend".

The keywords that may be used for the local host definition are:

1) **ip:** The IP version of the local host address
2) **address:** The local host address, which is the address that all secondaries must use to connect to this primary
3) **port:** The port secondaries use to connect to this primary (default = 5653)
4) **key_generation:** The method of generating random numbers (1 = software, 2 = hardware)
5) **rekey_wait:** The period in seconds to wait between rekeying (default = 3600)
6) **array_size:** The size of key arrays, if used (default = 256)
7) **data_array_time:** The period, in seconds, between using an index in the data key array instead of the actual key (default = 86400)
8) **control_array_time:** The period, in seconds, between using an index in the control key array instead of the actual key (default = 82800)
9) **heartbeat_time:** The time, in seconds, to wait before sending a heartbeat command (default = 15)
10) **heartbeat_fail:** The time, in seconds, to wait after a heartbeat command is sent before determining that the remote host has lost contact (default = 120)
11) **cluster_state:** Clustering definition flag is currently unsupported (default = 0)
12) **load_balance:** Load balancing definition flag is currently unsupported (default = 0)

There may be multiple remote host definitions. Each one is preceded by an identification number. All definitions with the same identification number are applied to the same remote host. The identification number is symbolized by "#" in the descriptions below.

The keywords that may be used for the remote host definition are:

1) **#/ip:** The IP version of the remote host address
2) **#/address:** The address of the remote host
3) **#/subnet#:** The subnet for which packets must be sent to this remote host. There may be multiple subnets, and they are differentiated by appending an identification number at the end of the *subnet* keyword
4) **#/mask#:** The mask of the subnet, with the same identification number appended to the *mask* keyword. If the remote host is a standalone $P^3$ host then the mask must be 255.255.255.255
5) **#/key_type:** The type of encryption to be used (default = AES128, alternative = AES256)
6) **#/rekey_wait:** Overrides the global rekey wait parameter in the local host definition above for this secondary
7) **#/key_array:** Overrides the global key array parameter in the local host definition above for this secondary
8) **#/data_array_time:** Overrides the global data array time parameter in the local host definition above for this secondary
9) **#/control_array_time:** Overrides the global control array time parameter in the local host definition above for this secondary
10) **#/heartbeat_time:** Overrides the global heartbeat time parameter in the local host definition above for this secondary
11) **#/heartbeat_fail:** Overrides the global heartbeat fail parameter in the local host definition above for this secondary

# 2 Secondary

The local section of the secondary configuration must begin with the keyword "localstart" and end with the keyword "localend".

The keywords that may be used for the local host definition are:

1) **ip:** The IP version of the local host address
2) **address:** The local host address, which is the address that all secondaries must use to connect to this primary

3) **cluster_state:** Clustering definition flag is currently unsupported (default = 0)
4) **load_balance:** Load balancing definition flag is currently unsupported (default = 0)

There may be multiple remote host definitions. Each one is preceded by an identification number. All definitions with the same identification number are applied to the same remote host. The identification number is symbolized by "#" in the descriptions below.

The keywords that may be used for the remote host definition are:

1) **#/ip:** The IP version of the remote host address
2) **#/address:** The address of the remote host which must match the address definition in the primary host
3) **#/subnet#:** The subnet for which packets must be sent to the remote host. There may be multiple subnets, and they are differentiated by appending an identification number at the end of the *subnet* keyword
4) **#/mask#:** The mask of the subnet, with the same identification number appended to the *mask* keyword. If the remote host is a standalone $P^3$ host then the mask must be 255.255.255.255
5) **#/port:** The port used to connect to the primary (default = 5653)