Global Information Assurance Certification Paper

CIPE: A Poor Man's VPN (Virtual Private Network)

Stacy Bresler

December 28, 2000

Introduction

Private data is intended to be kept private. How do we accomplish such a task without compromising the bottom line? The VPN commercial market would have you believe that you need to spend a lot of money to secure your data. What they don't tell you is that it is possible to do this with next to nothing. One VPN solution that can be accomplished with a minimal amount of expenditure is CIPE.

CIPE stands for Cryptographic IP Encapsulation. It was designed as a lightweight alternative to IPSEC (IP Security Protocol). The latter of which tends to carry a lot of "extras" that aren't always required for the simple VPN solution. [Note: IPSEC supplies a standards framework for securing IP traffic.][i]

This paper is designed to provide the reader with a rudimentary understanding of CIPE and a how-to guide in the installation and configuration of the components. I will attempt to provide a simplistic view of a technology that, at times, can be intimating.

Although CIPE is being ported to the Win32 environment, the port is still being stabilized and has many fixes yet to be implemented. In order to try to avoid any "catastrophic" downtime with one's operating system, I will be presenting a deployment based on the Linux version. If you are so bold to attempt an installation of the Win32 port, the code can be found here:

http://cipe-win32.sourceforge.net/

*NOTE: I have read of many people who have successfully implemented the current version of CIPE for Win32; however, it has been this writer's experience (and those of his testing cohorts at Foghorn Security) that the code stills needs a "little more **fixin'"** before I would recommend its usage.*

Background

So you ask, what is a VPN? According to the Virtual Private Networks FAQ compiled by Tina Bird, "…a VPN is a group of two or more computer systems, typically connected to a private network (a network built and maintained by an organization solely for its own use) with limited public network access, that communicates "securely" over a public network." [ii]

Securely is the operative word. It is how a VPN performs its security that defines the products' strengths and weaknesses. We can assume that the VPN will perform some

type of encryption and will deploy some method to use strong authentication to identify the parties that will participate in the VPN. Lastly, the VPN must hide information regarding the private networks that it is designed to protect.

Some practical uses of a VPN include remote employee access to a corporation's private network or a business-to-business (b2b) connection over the Internet with the purpose of transmitting confidential data. One other use might be a connection between a corporation's geographically separated offices that require network resources to be shared and a budget that only allows for a couple of ISDN lines to the local Internet Solution Provider.

CIPE is one of many free VPN solutions available today.

How CIPE works and other tidbits of information

This is a simple explanation and does not get into the nitty-gritty of CIPE's inner workings. If you want to delve into the details read Olf Titz's detailed documentation:

http://sites.inka.de/sites/bigred/devel/CIPE-Protocol.txt

Upon installation of CIPE, a new network interface (a "software" interface) is added. This interface allows CIPE to intercept the network stack. The result is that IP packets routed through this interface are encrypted and sent through a tunnel to a peer gateway. Once received at the other CIPE endpoint, the packets are decrypted and delivered to their destination. Keep in mind, CIPE "assumes a fixed link between two peers which exchange datagrams." [iii] Simply put, this is a peer to peer system where multiple interfaces are required for multiple tunnels.

A unique aspect of CIPE is that it uses UDP. The benefit of using UDP is speed (no delivery overhead) and the fact that the protocol is easily handled resulting in a firewall friendly environment.

An additional strength of CIPE is noted by Marc Mutz:

> *CIPE is a very simple package in that it requires you to make some decisions at compile-time, which leads to specialized and simpler code, which in turn is supposedly less bug-loaden than a complicated protocol like IPSec. CIPE is only compatible to itself, but often enough that is no limitation.* [iv]

The default cipher that is used for CIPE is Blowfish (more details on Blowfish are available here: http://www.counterpane.com/blowfish.html). There is also an option to use IDEA. Another one of CIPE's default settings is the use of a shared secret key. This is not the only option. As of v1.5, CIPE has introduced PKCIPE. PKCIPE is a public-key based configuration that does not rely on shared keys.

An interesting note regarding CIPE deployment platforms is that it is now available as a module for the LRP – The Linux Router Project. More information available at: http://lrp.c0wz.com/

Installing CIPE on Linux

The most common environment (and probably most useful for our simple explanation) is one where two private networks are connected to the Internet and each network has a properly configured Linux firewall in front of a private network.

Assumptions

For the purposes of this paper, let us assume that some basic Linux commands are understood. This installation will not utilize the PKCIPE module, instead we will provide a shared secret key.

For reference, the following details the two networks:

FIREWALL A	FIREWALL B
FWA ETH0: 24.10.65.1	FWB ETH0: 48.10.65.1
FWA ETH1: 10.1.1.1	FWB ETH1: 10.2.1.1
PRIVATE NETWORK: 10.1.0.0	PRIVATE NETWORK: 10.2.0.0

Prerequisites

You must have at a minimum the complete "include" tree of the running kernel installed. This is generally located at /usr/src/linux/ .

In order to make CIPE function, IP Forwarding will need to be enabled (this is because you must "forward" all packets requiring encryption to the CIPE network interface). Issuing the following command can do this; however, this most likely is enabled as part of your firewall configuration:

echo 1 > /proc/sys/net/ipv4/ip_forward

Obtaining the Code

First of all we must obtain the CIPE source code (for our purpose, we will assume that CIPE is not part of the base installation of Linux -- although it would appear that Linux 7.0 includes CIPE).

You can obtain the latest CIPE source code at:

http://sites.inka.de/sites/bigred/devel/cipe.html

Download the current stable release to the directory of your choice on the target Linux firewall. An installation will be required on both firewalls. I will be detailing the installation of one end of the tunnel, so simply repeat these steps on the other firewall (changing the configuration as required).

At this point you will need to decompress the tarball. The following command will accomplish this task:

tar xfz cipe-1.5.1.tar.gz

Compile
In the directory that you decompressed the tarball type (this assumes that the source of the kernel in /user/src/linux/):

./configure

If the kernel source is elsewhere you will have to tell CIPE where it is using the - -with-linux option:

./configure - -with-linux=/user/src/elsewhere

Assuming that the kernel source was available, there should be no errors. Be sure to pay particular attention to the output and reference the *cipe.info* file to help in troubleshooting.

Another configuration option is to enable the IDEA cipher rather than the default Blowfish. This is accomplished by providing the option - -enable-idea at the time of running *./configure*. Here's an example:

./configure - -enable-idea

Note that a new directory has been created. There is meaning to this directory and it is named according to some specifics. Here is an example of the directory name: 2.2.14-i386-ci . The **2.2.14** tells us the version of the kernel. **i386** lets us know the machine type. The next section tells us the protocol version and the cipher that was used – c is the protocol and i is for IDEA (b would be used for Blowfish). This directory naming is convenient for when you are compiling multiple versions of the CIPE package.

OK. We have made it this far and we are ready to compile. Change to the new directory and type:

make && make install

Put the Files in the right place

Copy the files from the **samples** directory to /etc/cipe/ . Make sure that the **options** file has the permissions of *rw* only for root.

Configuration

We need to edit the **options** file on each firewall. Below are the examples of the **options** file as it would apply to the aforementioned example environment. The Port number can be whatever you like. The ipaddr and ptpaddr are the IP addresses of the cipcb interface and the IP address of the peer's cipcb interface. You will want to create a key file that will be used for both systems. I suggest using *md5sum* to create the shared key:

md5sum -s somephrasehere

Firewall A options file:

----------cut here----------

device cipcb

IP address for CIPE interface on this system
ipaddr 192.168.65.10
IP address for remote CIPE interface
ptpaddr 192.168.65.9
Firewall B public IP:PORT
peer 48.10.65.1:9000
Firewall A public IP:PORT
me 24.10.65.1:9000
Shared key – this must be the both ends of the CIPE tunnel
key a8a2e5a07b1866c449e7db85c50bba6f

----------cut here----------

Firewall B options file:

----------cut here----------

device cipcb

IP address for CIPE interface on this system
ipaddr 192.168.65.9
IP address for remote CIPE interface
ptpaddr 192.168.65.10
Firewall A public IP:PORT
peer 24.10.65.1:9000
Firewall B public IP:PORT
me 24.10.65.1:9000
Shared key – this must be the both ends of the CIPE tunnel
key a8a2e5a07b1866c449e7db85c50bba6f

----------cut here----------

Start CIPE
Before we can start CIPE we must load the modules on both of the firewalls. To do this issue the following command:

modprobe cipcb

Finally, start CIPE:

ciped-cb

That's it folks! You can test the tunnel by pinging the peers **ptpaddr.** In our example you would ping 192.168.65.9 from firewall A and ping 192.168.65.10 from firewall B.

One final note - the most frequent problem that people encounter with CIPE is that they forget to update their firewall rules not to block the UDP traffic required.

For more information on CIPE and masquerading be sure to review the notes here: http://www.linuxdoc.org/HOWTO/mini/Cipe+Masq.html . When reading the notes at this site take note of their warning:

> *"The ip-up scripts currently only allow class c traffic through the cipe interface. If you wish for machine B to communicate with Machine C then you will need to change the appropriate ip-up and ip-down scripts."*[v]

Conclusion

In this practical I have introduced you to CIPE. Granted there are numerous other free VPN solutions available. For the curious VPN explorer you might want to venture to these websites for more information:

http://www.crosswinds.net/nuremberg/~anstein/unix/vpnd.html

http://www.worldvisions.ca/tunnelv/

http://www.flora.org/freeswan/

http://poptop.lineo.com/

http://www.antd.nist.gov/cerberus/

These free tools do not claim to offer all the features that a commercial VPN product promotes. However, for minimal effort and minimal cost CIPE (and its other no cost counterparts) can be deployed to meet your need to secure data over unsecured channels.

[i] Norberg, Stefan. "Securing Windows NT/2000 Servers for the Internet." January 2001. p. 87

[ii] Bird, Tina. "Virtual Private Networks Frequently Asked Questions." 4 May 2000. URL: http://kubarb.phsx.ukans.edu/~tbird/vpn/FAQ.html (29 December 2000).

[iii] Titz, Olaf. "CIPE-Protocol." 30 May 1997. URL: http://sites.inka.de/sites/bigred/devel/CIPE-Protocol.txt. (15 January 2001).

[iv] Mutz, Marc. "Linux Encryption HOWTO." v0.2.0. 24 February 2000. URL: http://thibs.menloschool.org/help/encryption/Encryption-HOWTO-5.html (15 January 2001).

[v] Ciaravalo, Anthony. "The Linux Cipe+Masquerading mini-HOWTO." v1.2. 21 April 1999. URL: http://www.linuxdoc.org/HOWTO/mini/Cipe+Masq.html

Last Updated: May 26th, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event