Enova Technology Whitepaper

Enigma Product Whitepaper
Securing Data at Rest Safely and Easily

Introduction

Today more than ever before, individuals and Corporations of all sizes wrestle with how to best protect sensitive and confidential data. Compound this with the fact that users of all types have more data to protect – and the data acquisition trend continues to rise.

One of the core problems that have historically plague consumers and IT organizations alike has been the need to securely protect confidential data, be it the data-at-rest or data-in-transit. Due to recent legislation, it is now incumbent upon the Global Corporate Community to secure data of all types. The ongoing of "how to best accomplish this goal" is the $64 Billion question.

The tight linkage between the multiple software and hardware solutions available to the user often frustrates optimization efforts, as it is difficult or impossible to provision and de-commission all security solutions in line with constantly changing business environment.

As a result, businesses constantly face the delicate balancing act of allocating too few hardware resources and risking data losses, or potentially wasting resources and money by using too many solutions.

While people consider notebooks, laptop and portable storage devices to be very valuable equipment, they don't think about the implications of the theft until it's too late. Businesses, especially, should consider that the data stored on mobile storage devices is almost certainly much more valuable than the hardware, which can be replaced quickly. A windows password or encrypted zip files won't do the trick, as thieves will typically have access to most data once they access the storage drive on another system.

Every IT product available makes a claim as to functionality and/or offered security. When protection of information and communications used in e-commerce, critical infrastructure and other sensitive stored data, Federal agencies (regulated by extensive legislative restrictions) and Enterprises need to know that a product's stated security claim is valid.

At the core of all cryptographic product offerings is the cryptographic module, which offers services such as data encryption and authentication. FIPS 140-1/2 validates the cryptographic module and its underlying cryptographic algorithms against established standards to fend off any weakness and loopholes of a design. Level 2 requirements of 140-2 certification include the ability to provide Tamper-evident physical security, as well as role-based authentication.

Over the past 15 or so years, numerous data protection solutions have been presented in both hardware and software. Along the way, Users (which consist of Consumers, SMB, Enterprise and large Institutional Enterprise firms, such as Government, Finance, Insurance and Medical verticals) have had their PCs, Laptop or mobile drives lost or stolen; rendering all of the data free and easily accessible to the new owner.

Introduced over the past 15 years, no fewer than 6 major pieces of legislation (California's SB1386, DoD 5015.2-STD, GLBA, HIPPA, SOX, and the Data Protection Act) require that all Data-at-Rest (DAR) be hardware protected using AES-256 encryption (using either EBC or CBC modes). These stringent DAR requirements are mandatory to protect consumer privacy, offer strong encryption, protect against data alteration or destruction, and employee privacy data.

To make an informed selection from the multitudes of devices – both software and hardware alike – is a formidable task today.

However, one new data encryption/decryption protection solution offered by Enova Technology makes the choice much simpler and easier to the User. Enova's new product, ***Enigma***, offers the User a very sophisticated yet simplistic approach to full disk encryption (FDE) data protection. While this generation addresses only full disk encryption, more solutions will be introduced shortly to facilitate the Data-in-Transit encryption applications, especially in Cloud Computing.

How will Enigma solve the privacy & confidentiality issues surrounding mobile data?

The *Enigma* module is a USB real-time full disk encryption device, designed to transparently encrypt all data on any number of USB based storage devices including those USB1.0, USB2.0 and USB3.0 enabled drives. The *Enigma* module is operating system independent, making it a perfect choice for various system platforms (Windows[1], Mac, Linux and Android) with a standard USB protocol support.

- ✓ USB thumb drives
- ✓ USB hard drives (including the new hybrid SSD/HDD drives)
- ✓ USB backup drives
- ✓ USB Blu-Ray and DVD storage DVD and CDs
- ✓ USB card readers

Figures 1-3, from left clockwise: Photo of *Enigma* module; the *Enigma connected to a USB3.0/SATA hard drive;* the *Enigma* connected to a USB2.0 Thumb drive.

Features and Benefits

- NIST/CSE certified hardware AES ECB and CBC hardware crypto engine up to 256-bit strength
- Automatically and transparently real-time encrypts ***<u>any number</u>*** of USB based storage devices
- Support all major OS including Windows, MAC, Linux and Android
- Compatible with USB1.0, USB2.0 and USB3.0

[1] Windows and Macintosh are registered marks of Microsoft Corporation and Apple Computers respectively.

- Key Management through the Recovery Password that allows the user to regain access to encrypted data in the event that your *Enigma* is lost, stolen, or may have malfunctioned
- NO software or drivers to install or learn
- No slowdown or delay due to the encryption process; performance is actually limited by the storage media's read/write speeds
- Leaves no trace over the host computer
- Manufacturer's suggested retail is $49.95 in the US markets for *Enigma* AES ECB 256-bit model
- Manufacturer's suggested retail is $79.95 in the US markets for *Enigma* AES CBC 256-bit model

Instant Installation and Usage of Enigma

The *Enigma* module, which acts as a USB real-time encryption bridge, is simple to use, requiring only 2 clicks to get started after removing it from the package!

Simply connect the USB drive to the *Enigma* module then connect the pair (*Enigma* and USB drive) to your host computer USB port to start the 2 steps initialization process. After successful initialization, the *Enigma* module automatically and transparent encrypts/decrypts all data written to/read from the connected USB drive.

To demonstrate, connect your newly encrypted USB device - without the *Enigma* module attached – and note that your computer will not detect any data on that encrypted USB drive. Instead, the computer will prompt a warning message that indicates the connected USB drive is a new drive requiring formatting. That is because the entire drive is encrypted by the *Enigma* module.

Conventions

USB MSC – USB Mass Storage Class; In general, it refers to a storage device under USB protocol;

AES - Advanced Encryption Standard as published in FIPS 197;

ECB - Electronics Code Book, is a confidentiality mode that features, for a given DEK, the assignment of a fixed cipher text block to each plaintext block, analogous to the assignment of code words in a codebook. Essentially the same DEK is applied to every plaintext data block independently.

CBC - Cipher Block Chaining, is a confidentiality mode whose encryption process features the combining ("chaining") of the plaintext blocks with the previous cipher text blocks. The CBC mode requires an IV (Initialization Vector) to combine with the first plaintext block, in addition to using a given DEK. The security level of a CBC implementation is quadruple trillions times more than that of an ECB.

DEK - Data Encryption Key, a 256-bit key responsible for data encryption and decryption based on AES ECB or CBC mode of operation.

Default DEK - Default Data Encryption Key, a 256-bit key embedded during the manufacturing process. This default DEK can be changed or replaced upon entering the "Recovery Password."

Enigma_CD – an Enigma on-board emulated CD-ROM that contains all software code utilities including the main utility "enigma.exe," quick guide, user's guide, warranty and FAQ.

Initialization - a process required to initialize the *Enigma* dongle by asking the user to enter and confirm the "Recovery Password," which allows the user to regain access to encrypted data in the event that *Enigma* is lost, stolen, or may have malfunctioned. If the *Enigma* dongle wasn't properly initialized, the

Enigma connected USB drive won't be able to be formatted. A system warning message of “This device is write-protected.” will prompt when a user performs the action of “FORMAT” through an un-initialized Enigma dongle.

Recovery Password – an essential part of the *Enigma* key management. The recovery password is responsible for generating the DEK value to regain access to the encrypted data, in the event that your *Enigma* is lost, stolen, or may have malfunctioned. The same recovery password generates the same DEK. This “Recovery Password” should be kept with confidentiality.

Start Programming - a part of the “Recovery Password” process that writes the new DEK value to the *Enigma* dongle to replace the old DEK value. Note that a new DEK value can be equivalent to the old DEK value.

Write-Protect Entire Storage – Using *Enigma*, the **“*write-protect entire storage*”** feature disables all write operations to the connected USB drive, making the USB drive “read-only” which effectively rejects any virus, spyware or malware intrusion. While this feature is turned ON, all contents of the drive remain encrypted. There isn’t any extra driver to install. See also **“*Safeguard Boot Sector.*”**

Safeguard Boot Sector[2] – Using *Enigma*, the ***“Safeguard Boot Sector”*** feature disables all write operations to the boot sector of a connected USB thumb drive and card reader storage media, making it “read-only” which effectively rejects any virus, spyware or malware intrusion. While this feature is turned ON, write operation is only permitted to the other sectors with written data being encrypted. More, the *Enigma* blocks the “autorun.ini” type malware so that the malware won’t be able to implant itself to the *Enigma* built-in CD-ROM (***Enigma_CD***) that contains all software code utilities. There isn’t any extra driver to install. See also “***Write-Protect Entire Storage.***”

Firmware Update – a process that replaces your existing firmware using a newer version. This is only required whenever there is a newer version release that resolves known compatibility issue or enhances the security operation. There isn’t any extra driver to install.

Initialize the Enigma real-time full disk encryption dongle

The *Enigma* module must be initialized to enable its full functionalities. To initialize, follow these three simple steps:

1. Insert the USB drive to the female connector of the *Enigma* module;
2. Insert the *pair* to any host USB port and ignore or cancel Windows’ “Format” prompt. The system will automatically detect a CD-ROM labeled as “***Enigma_CD***.” Use Windows Explore (My Computer) to open the “***Enigma_CD***.” To initialize, execute “***enigma.exe***” as found in the ***Enigma_CD*** content. The code utility will guide you through the 2-step initialization procedure. At the completion of the initialization, the code utility requires you to unplug then re-plug the pair to allow changes to take effect;
3. At the re-plug, OS detects the attached USB drive, and asks if you want to format the drive. Click “Yes” and finish the formatting. Now your USB drive is associated with the new Data Encryption Key (DEK) of the *Enigm*a module. You can start operating the USB drive just like operating a regular USB drive except that all data written to it are automatically transparently encrypted.

[2] Using *Enigma*, the “Safeguard Boot Sector” feature also protects the Master Boot Record (MBR) of a connected USB/SATA type fixed disk drive.

Enova Technology Whitepaper

Various Application Scenarios

Scenario #1: Creating a new DEK via setting up the "Recovery Password[3]" from the code utility

– this is a very first step that a user should perform by establishing the "Recovery Password" in the case of *Enigma* module lost, stolen or may have malfunctioned. Keep the "Recovery Password" with confidentiality. Note that same "Recovery Password" generates the same DEK that would allow you to regain access to your encrypted drive.

1. Connect the *Enigma* to the USB storage device;
2. Insert the *Enigma* into one of the USB ports on your computer;
3. Initialize the *Enigma*: this process asks the User to enter a Recovery Password. Re-enter the recovery password to assure its correctness, then click OK. You will see a "Programming Successful - Please unplug then re-plug the *Enigma* with the USB drive to save all changes" appears, simply follow the directions;
4. Once you have created this new DEK and you have plugged the *Enigma* and the USB drive back in to the USB port on the PC, a window will appear asking if you would like to format the drive;
5. Format the drive then start using it just like using a regular USB drive;
6. When you have completed your work, remove the connected pair (Enigma and Storage Device).

Scenario #2: Using the authentication process as a means of authentication

– this is the basic operation of the *Enigma.* The encrypted USB drive will always present itself to the computer as an unformatted media without the *Enigma* module connects to it. This is considered as a single factor authentication – that encrypted USB disk needs to connect to the *Enigma* to start data read/write.

1. Connect the encrypted USB drive to the computer *without* connecting the *Enigma*;
2. A window will appear, asking if you would like to format your device (as the computer cannot 'see' any files or data, it presumes that this is a new storage device);
3. Remove the USB drive;
4. Plug the USB drive back into the *Enigma*;
5. Plug the *Enigma* with the drive into the USB port;
6. A window will appear asking if you would like to Open Folder to View your Files;
7. Click on Open folder to view your files; you may add more files, or delete files at your discretion.

Scenario #3: Using the "Recovery Password" as a means for 2-factor[4] authentication

– one other important function of the "Recovery Password" is to use it as a means for 2-factor authentication (something you have which is the *Enigma* and something you know which is the "Recovery Password.") At the completion of every *Enigma* usage, simply reset the "Recovery Password" through the code utility (enigma.exe) as example shown below. You must memorize the "Recovery Password" and enter the correct one upon your next usage however.

1. Plug the USB drive into the *Enigma*;
2. Plug the *Enigma* into the host USB port;
3. Note a window will appear asking if you would like to Open Folder to View your Files;
4. Click on Open folder to view your files; you may add more files, or delete files at your discretion;

[3] The "***Recovery Password***" is engineered specifically to facilitate simple yet effective key management. It is established at the initialization stage to generate a Data Encryption Key (DEK) to operate on USB drive. Same Recovery Password generates same DEK that would allow you to regain access to the encrypted USB drive.

[4] 2-factor authentication is a lot more secure than the single factor. In this application, *Enigma* is something you have (first factor) and that something you know (the Recovery Password) is the 2nd factor.

5. At the end of your operation, use the code utility (enigma.exe) to reset your recovery password back to, for example, six zeros (000000);
6. Unplug then re-plug to save new settings. Now your *Enigma* has been programmed with a different DEK, so that you can safely place it along with the encrypted USB drive;
7. When trying to regain access to the encrypted USB drive, use the same code utility (enigma.exe) to set the correct recovery password. Unplug then re-plug to save new settings. Now you can regain access to your encrypted USB drive.

Scenario #4: Using the recovery password as a means for 2-factor authentication for traveling

– for those road warriors who travel with confidential USB mobile data, resetting “Recovery Password” is highly recommended as by doing so, you can place the *Enigma* and the encrypted USB drive all together without worrying about losing the pair. You must memorize the “Recovery Password” and enter the correct one upon your next usage however.

1. Plug the USB drive into the *Enigma*;
2. Plug the *Enigma* into the host USB port;
3. To regain access to the encrypted USB drive, use the same code utility (enigma.exe) to set the correct recovery password. Unplug then re-plug the pair to save new settings;
4. At the re-plug, a window will appear asking if you would like to <u>Open Folder to View your Files;</u>
5. Click on Open folder to view your files; you may add more files, or delete files at your discretion;
6. At the end of your operation, use the code utility (enigma.exe) to reset your recovery password back to, for example, six zeros (000000);
7. Unplug then re-plug to save new settings. Now your *Enigma* has been programmed with a different DEK, so that you can safely place it along with the encrypted USB drive for traveling;
8. When trying to regain access to the encrypted USB drive, use the same code utility (enigma.exe) to set the correct recovery password. Unplug then re-plug to save new settings. Now you can regain access to your encrypted USB drive.

Scenario #5: Using the recovery password as a means to access the same encrypted data on an encrypted USB drive in a different geographic location

– one or multiple remote offices that need access to the encrypted USB drive received and each office has equipped with at least one *Enigma*.

1. Connect the *Enigma* to the encrypted USB drive;
2. Call the office and ask for the recovery password that’s responsible for encrypting the USB drive you now have;
3. Set the correct recovery password through the code utility (enigma.exe); Unplug then re-plug the pair to save the settings;
4. In a few seconds, the computer will generate the now familiar window, asking if you would like to <u>Open Folder to View your Files;</u>
5. Simply open the folder and show your data files.

(Connect the USB storage device itself without connecting it to the *Enigma*, and you by now already know what message will appear on the screen - the computer will ask if you would like to format the new drive).

Scenario #6: I’ve lost my Enigma dongle - now what?

– the “Recovery Password” you had established at initialization would allow you to regain access to your encrypted USB drive in the case of lost or stolen. Simply go out and purchase another *Enigma* and initialize it with the same “Recovery Password” of record.

1. Go to any store or order another *Enigma*;
2. Once you have opened the new *Enigma* product, insert the encrypted USB drive and then insert the pair into the computer;
3. Initialize as before; be sure to input the original Recovery Password of record;
4. Follow the steps, then unplug the pair and re-plug to save all changes;
5. Upon re-plug, you will see a window asking if you would like to Open Folder to View your Files;
6. Click on Open folder to view your files; you may add more files, or delete files at your discretion.

Scenario #7: Daisy-Chained multiple Enigma modules to perform key escrow of an encrypted USB drive

– for security paranoids, two or more *Enigma* can be "daisy-chained" together to create an unprecedented level of data security – with double or treble AES 256-bit encryption without losing performance. The *Enigma* encryption sequence matters in the daisy-chained operation and you can practice that as long as your computer bus power[5] permits.

1. Ensure that both Enigma are properly initialized;
2. Connect the USB drive to the first *Enigma* then cascade the 2nd Enigma dongle;
3. Connect the trio to the host USB port then format the USB drive. Now the USB drive is encrypted by two different DEKs. The sequence of Enigma matters when you try to decrypt the USB drive;
4. Assign the two Enigma to two responsible individuals;

Scenario #8: I would like to use a USB drive which already has data on it. What's the process to do this without losing my data that is already on the drive?

– You'd like to preserve and encrypt a USB drive with data on it before the drive being formatted by the *Enigma*. The simplest way to accomplish this is to make a copy of the data to your desktop or another location, then connecting that drive to the *Enigma*, and follow the initialization instructions. Once you've created a recovery password, drag the data back to the USB disk through the *Enigma* to encrypt all of the data.

1. Connect the USB drive alone to the PC;
2. Create a new folder on your desktop then drag and drop all data on the USB drive to the folder;
3. Remove the USB drive and connect it to the *Enigma*;
4. Perform the initialization procedure of *Enigma*; at the completion, unplug then re-plug the pair to save the changes;
5. At the re-plug, you will be asked to format the USB disk; you may now do so;
6. When the format is completed, you can start using your USB disk;
7. Drag and drop the content of the folder on your desktop to the USB disk through *Enigma*. All data written will be automatically transparently encrypted.

[5] The *Enigma*, like your USB drive, draws power directly from the host USB bus which has limited power supply. You will need to ensure the level of daisy-chained to not cause any negative effect to the data operation with special caution to the use of USB/SATA hard disk drives that draw power directly from the USB host.

Scenario #9: I have some used SATA and IDE[6] drives. Do I get to use the Enigma to encrypt[7] those drives?

– for sure you'd want to utilize additional storage sitting around. You may have some used SATA or IDE disk drives and you'd want them to be encrypted by the *Enigma.* Please back up all data to your PC or Network prior to beginning the operation outlined below.

1. Go out and purchase the USB to SATA and USB to IDE enclosures (without disk drives in it);
2. Install your disk drive then plug the USB drive to *Enigma*;
3. Plug the *Enigma* to the host USB port;
4. Assuming you had done initialization of the *Enigma*, at the plug in, the host will detect a USB disk and ask you to format it; Click "Yes" to format the drive;
5. When the format is completed, you can start using your USB disk;
6. Drag and drop the content of the folder on your desktop to the USB disk. All data written will be automatically transparently encrypted.

Scenario #10: Creating a "Write-Protect" USB drive from the code utility

– the Write-Protect feature enables the "Read Only" of the connected USB storage device through *Enigma*. Only the read operation is carried out whereas all stored data remain encrypted. Any write operation to the USB storage device will not be permitted. The function is particularly useful when it comes down to legal/law enforcement as evidence presented by a forensic created disk drive can not be over-written. Another application may be that organization may want to have the USB drive content as "read-only" to avoid virus and malware contamination when it is outside of a controlled environment. Additional application may be that corporate IT Help Desk[8] utilizes to fend off malicious virus/worm while keep intelligent properties safe and secure.

1. Connect the *Enigma* dongle to the USB storage device;
2. Insert the pair into one of the USB ports on your computer;
3. Initialize the *Enigma* dongle: this process asks the User to enter a Recovery Password. Re-enter the recovery password to assure its correctness, then click OK. You will see a "Programming Successful - Please unplug then re-plug the *Enigma* dongle with the USB drive to save all changes" appears, simply follow the directions;
4. Once you have created this new DEK and you have re-plugged the pair back in to the USB port on the computer, a window will appear asking if you would like to format the drive;
5. Format the drive then start using it just like using a regular USB drive;
6. Execute the same code utility (enigma.exe) and select "Anti-Malware;"

[6] After your purchase of the USB/SATA or USB/IDE bridged enclosure, install the drive into it and perform a simple test to see if the product works under a regular USB host port. If Windows successfully detects that drive, the product works. This is to ensure that your purchase is compatible with your host computer prior to connecting to an *Enigma* dongle.

[7] Even the *Enigma* dongle can start encrypting all written data automatically, depending on the used drive geometry, there may be plain text data previously written which are not protected. The file level "FORMAT" command simply won't erase all your previously written data as Forensic tools are still able to read out raw data. To ensure full protection of your used drive, copy multiple GB data several times to the USB drive through the *Enigma* to over write the plain text data if possible.

[8] **Tony Kiser, CISSP/SME - "not only will home users benefit from the enigma, but enterprise support will also benefit. With the growth of malware, APTs, and targeted attacks, so many tools are needed by support personnel... The *Enigma* will allow support techs to carry all the necessary tools to fight malware, and those tools will stay protected. In addition to this, the protection of intellectual property can be greatly increased with the use of the multi-point authentication the *Enigma* offers. Great product."**

7. Check on “Write-Protect Entire Storage” to enable[9] the write protection feature; you must remove the pair then re-plug for the changes to take effect.

Scenario #11: Safeguard the boot sector of a USB drive to fend off the boot sector virus

– The “Safeguard boot sector” feature prevents malware/virus from infecting to the boot sector of a connected USB thumb drive or USB card reader storage media through *Enigma* module. When this protection is turned on, you are permitted to read/write data with boot sector being protected. All data written remain encrypted. Different from the scenario #10 that creates a “read only” USB disk, organizations may want to allow the user’s data read/write with boot sector being protected.

1. Connect the *Enigma* dongle to the USB storage device;
2. Insert the pair into one of the USB ports on your computer;
3. Initialize the *Enigma* dongle: this process asks the User to enter a Recovery Password. Re-enter the recovery password to assure its correctness, then click OK. You will see a ”Programming Successful - Please unplug then re-plug the *Enigma* dongle with the USB drive to save all changes” appears, simply follow the directions;
4. Once you have created this new DEK and you have re-plugged the pair back in to the USB port on the computer, a window will appear asking if you would like to format the drive;
5. Format the drive then start using it just like using a regular USB drive;
6. Execute the same code utility (enigma.exe) and select “Anti-Malware” tab;
7. Check on “Safeguard boot sector” to enable[10] the feature; you must remove the pair then re-plug for the changes to take effect.

Summary

Enova’s *Enigma* product is very simple to use yet it provides robust security solution for all USB mobile data. The key management is simple and thorough and for some specific Enterprise applications, the backbone of a network version key server has been done which can be integrated into some known key server software for easy migration.

Fortunately, tools like *Enigma* are blazing the trail towards a new data protection architecture that will deliver on the paradoxical needs defined in the Enova Technology messaging: superior performance, true reliability, and scalability combined with increased efficiency, simplicity in use, reduced cost and reduced complexities.

About Enova Technology (Enova)

Enova builds and provides comprehensive enterprise based security products for Data at Risk (DAR) management and compliance solutions on a global basis. Enova offers the tools to automate and protect data distributed to virtually any USB enabled storage device, and manages these efforts using simple to understand processes.

[9] Check the “***Write-Protect Entire Storage***” box to enable the feature and you must remove and re-plug the *Enigma* dongle for the changes to take effect. Similarly, un-checking the same “***Write-Protect Entire Storage***” box would require unplug then re-plug the *Enigma* dongle.

[10] Check the “***Safeguard boot sector***” box to enable the feature and you must remove and re-plug the *Enigma* dongle for the changes to take effect. Similarly, un-checking the same “***Safeguard boot sector***” box would require unplug then re-plug the *Enigma* dongle.

Using Enova data security products, security teams will become more proactive, while compliance teams can more effectively manage, collaborate and enforce secure accountability.

Enova's growing customer base includes leading Governments, Global 2000 organizations in the financial services, healthcare, retail, energy and utility, transportation and manufacturing. For more information, please go to www.enovatech.com or send email to info@enovatech.com.

You can also contact Mr. Robert Fleming at +1 303 915 4164 or email bob.fleming@enovatech.com.

Reference Materials

About NIST Cryptographic Algorithm Validation Program (CAVP)

The Cryptographic Algorithm Validation Program (CAVP, http://www.nist.gov/itl/csd/sma/cavp.cfm) encompasses validation testing for FIPS-approved and NIST recommended cryptographic algorithms.

The Cryptographic algorithm validation process is a prerequisite to the Cryptographic Module Validation Program (CMVP, http://www.nist.gov/itl/csd/sma/cmvp.cfm). The CAVP was established by NIST and the Communications Security Establishment Canada (CSEC) in July 1995.

The goal of the CAVP is to provide federal agencies – in the United States, Canada, and the United Kingdom – which confidence that a validated cryptographic algorithm, such as AES or TDES, has been implemented correctly. This is accomplished by designing and developing validation test suites to verify the correct implementation. Federal agencies, industry, and the public may now select certified implementations from the associated Algorithm Validation Lists and have confidence in the claimed level of security.

Enova Technology Products	Type of Algorithm	Date Approved	Certificate #
X-Wall LXE/XOE/MXE	AES EBC (128/192/256-bit)	Mar. 6, 2003	60
X-wall DX/SE/LX/XO	TDES 128/192-bit	Jan. 4, 2002	92
X-Wall FX/MX/DX	AES CBC (128/192/256-bit)	April 6, 2005	250

About the Cryptographic Module Validation Program (CMVP)

Unlike CAVP that validates a specific cryptographic algorithm, the CMVP validates the entire cryptographic module to Federal Information Processing Standards (FIPS) 140-1 *Security Requirements for Cryptographic Modules*, and other FIPS cryptography based standards. FIPS 140-2 was released on May 25, 2001 to supersede FIPS 140-1.

Cryptographic Modules validated as conforming to FIPS 140-1 and FIPS 140-2 are accepted by the Federal Agencies for the protection of sensitive information.

Every IT product available makes a claim as to functionality and/or offered security. When protection of information and communications used in e-commerce, critical infrastructure and other sensitive stored data, Federal agencies, regulated by legislative restrictions, need to know that a product's stated security claim is valid. At the core of all cryptographic products offering is the cryptographic module, which offers services such as data encryption and authentication. FIPS 140-2 validates the cryptographic module and its underlining cryptographic algorithms against established standards to fend off any weakness and loopholes of a design.

Enova Technology Crypto Modules	Type of Validation	Date Approved	Certificate #
X-Wall MX-256 single chip	FIPS 140-2	Dec. 28, 2010	1471
X-Wall MX-256C single chip	FIPS 140-2	Dec. 28, 2010	1472