proprivacy.com

AES Encryption | Everything you need to know about AES

Written by Douglas Crawford

18-23 minutes

AES is a symmetric key encryption cipher, and it is generally regarded as the "gold standard" for encrypting data.

AES is NIST-certified and is used by the US government for protecting "secure" data, which has led to a more general adoption of AES as the standard symmetric key cipher of choice by just about everyone. It is an open standard that is free to use for any public, private, commercial, or non-commercial use.

An introduction to AES encryption

AES is a symmetric key encryption cipher. This means that the same key used to encrypt the data is used to decrypt it. This does create a problem: how do you send the key in a secure way?

Asymmetric encryption systems solve this problem by securing data using a public key which is made available to everyone. It can only be decrypted by an intended recipient who holds the correct private key.

This makes asymmetric encryption much better at securing data in transit as the sender does not need to know the recipient's

private key. A good example is RSA encryption, which is used to secure the TLS key exchanges required when connecting to a secure HTTPS website.

Symmetric ciphers like AES are therefore much better at securing data while at rest – such as when it is stored on your hard drive. For this purpose, they are superior to asymmetric ciphers because:

- They require much less computational power. This makes encrypting and decrying data with symmetric encryption much faster than with asymmetric encryption. For perspective, symmetric ciphers are generally quoted as being around "1000 times faster" than asymmetric ones.
- And because they are faster, symmetric ciphers are much more useful for bulk encrypting large amounts of data. Asymmetric ciphers such as RSA are only really used for encrypting small amounts of data, such as the keys used to secure symmetric key encryption.

  Of course, in today's connected world, data that just sits on your hard drive is of limited use. Fortunately, it can be safely transferred over the internet in conjunction with asymmetric encryption, which used to handle the remote key exchanges required to securely connect to a remote server.

  OpenVPN, for example, secures the raw data with a symmetric cipher – usually AES these days. In order to transfer the encrypted data securely between your PC and the VPN server, it uses an asymmetric TLS key exchange to negotiate a secure connection to the server.

Is AES encryption the best type of encryption?

AES is widely regarded as the most secure symmetric key encryption cipher yet invented. Other symmetric key ciphers that are considered to be highly secure also exist, such as Twofish, which was co-invented by renowned cryptographer Bruce Schneier.

Such ciphers have not been battle-tested in the way that AES has, though. And hey, if the US government thinks AES is the best cipher to protect its "secure" data, who's arguing? There are some, however, who see this as a problem. Please see the section on NIST below.

Widespread adoption has benefited AES in other ways. Most CPU manufacturers have now integrated the AES instruction set into their processors. The hardware boost improves AES performance on many devices as well as improving their resistance to side-channel attacks.

Can 128-bit AES encryption be broken?

AES itself is unbreakable when implemented properly.

In 2011 the fastest supercomputer in the word was the Fujitsu K. This was capable of an Rmax peak speed of 10.51 petaflops. Based on this figure, it would take Fujitsu K 1.02 x 10^18 - around one billion billion (one quintillion) - years to crack a 128-bit AES key by force. This is older than the age of the universe (13.75 billion years).

The most powerful supercomputer in the world in 2017 was the Sunway TaihuLight in China. This beast is capable of a peak speed of 93.02 petaflops. This means that the most powerful computer in the world would still take some 885 quadrillion years to brute force a 128-bit AES key.

The number of operations required to brute force a 256-bit

cipher is 3.31 x 10^56. This is roughly equal to the number of atoms in the universe!

Back in 2011, cryptography researchers identified a weakness in AES that allowed them to crack the algorithm four times faster than was possible previously. But as one of the researchers noted at the time:

"To put this into perspective: on a trillion machines, that each could test a billion keys per second, it would take more than two billion years to recover an AES-128 key."

In response to this attack, an additional four rounds (see later) were added to the AES-128 encryption process to increase its safety margin.

Side Channel attacks

So to all intents and purposes, AES itself is unbreakable when implemented properly. But it not always implemented properly.

Side-channel attacks look for clues from the computer system implementing the AES encryption in order to find out additional information. This may be useful in reducing the number of possible combinations required to brute force AES.

These attacks use timing information (how long it takes the computer to perform computations), electromagnetic leaks, audio clues, and even optical clues picked up using a high resolution camera to discover extra information about how the system is processing the AES encryption.

A well-known side-channel attack against AES successfully deduced AES-128 encryption keys by carefully monitoring the cipher's shared use of the processors' cache tables.

Properly implemented AES mitigates against side-channel

attacks by preventing possible ways data can leak (which is where use of the hardware-based AES instruction set helps) and by using randomization techniques to eliminate the relationship between data protected by the cipher and any leaked data that could be collected using a side-channel attack.

Insecure Passwords

AES encryption is only as secure as its key. These keys are invariable themselves secured using passwords, and we all know how terrible us humans are at using secure passwords. Keyloggers introduced by viruses, social engineering attacks, and suchlike, can also be effective ways to compromise the passwords which secure AES keys.

Use of password managers greatly mitigates against this problem, as does use of two-way firewalls, good antivirus software, and greater education about security issues.

A brief history of AES encryption

When you were a kid, did you play the game in which you created a "secret message" by substituting one letter of the message with another? The substitution was made according to a formula picked by you.

You might, for example, have substituted each letter of the original message with one three letters behind it in the alphabet. If anyone else knew what this formula was, or was able to work it out, then they would be able to read your "secret message."

In cryptography jargon, what you were doing was "encrypting" the message (data) according to a very simple mathematical algorithm.

Encryption has been used hide to sensitive data since ancient

times, but really came in its own during the Twentieth Century. During World War 2 the Germans famously secured their communications using the Enigma machine, the code for which was equally famously cracked by Alan Turing at Bletchley Park.

What is DES encryption

The Data Encryption Standard (DES) was created in the mid-1970s to secure US government communications. It became the first modern, public, freely available encryption algorithm, and as such almost single-handedly created the modern discipline of cryptography.

Although developed by IBM, DES was the brainchild of National Bureau of Standards (NBS, which later became NIST).

Despite concerns about meddling by the NSA, DES was adopted by the US government in 1976 for "sensitive but unclassified" traffic. This included things like personal, financial and logistical information.

Since there was nothing else like it at the time, it quickly became widely adopted by commercial companies who required encryption to secure their data. As such, DES (which used 56-bit keys) became the default workhorse encryption standard for almost two decades.

This almost ubiquitous adoption was greatly helped by DES being awarded Federal Information Processing Standards (FIPS) status. All US non-military government agencies and civilian government contractors are required to use FIPS standards only.

By the mid-1990s, however, DES beginning to show its age. At this time it was widely believed that the NSA could brute-force crack DES, a point proved in 1998 when a $220,000 machine

built by the Electronic Frontier Foundation (EFF) successfully brute-forced DES in just two days. It was clearly time for a new standard.

How AES came about

In 1997 the National Institute of Standards and Technology of the United States (NIST) announced that was looking for a replacement to DES. In November 2001 it announced that the winner: AES, formerly known as Rijndael after one of its co-creators.

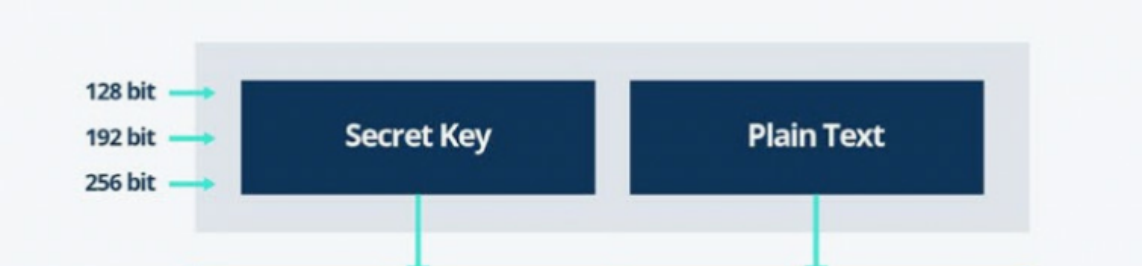
On NIST's recommendation, the new cipher was formally adopted by the US federal government and came into effective use in May 2002. Like DES before it, AES was awarded FIPS status. The US government considers all AES key sizes to be sufficient for classified information up to the "Secret" level, with "Top Secret" information requiring AES-192 or AES-256.

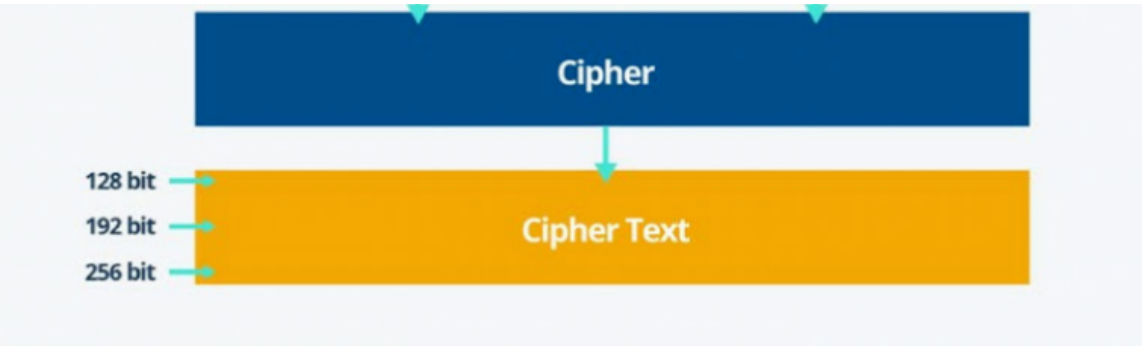
AES has now entirely replaced DES worldwide as the default workhorse symmetric encryption standard.

How does AES encryption work?

The AES encryption algorithm encrypts and decrypts data in blocks of 128 bits. It can do this using 128-bit, 192-bit, or 256-bit keys. AES using 128-bit keys is often referred to as AES-128, and so on.

The following diagram provides a simplified overview of the AES process…





Plain text

This is the sensitive data that you wish to encrypt.

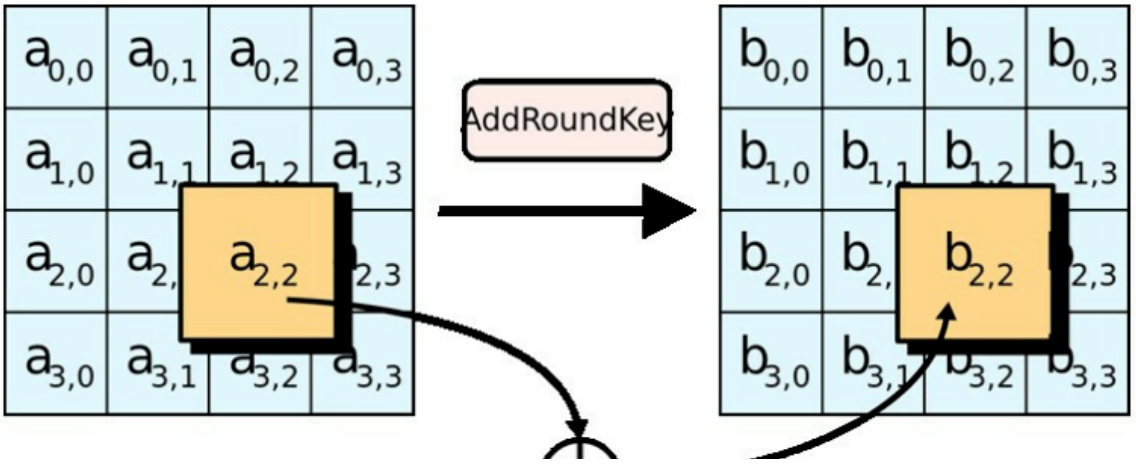
Secret Key

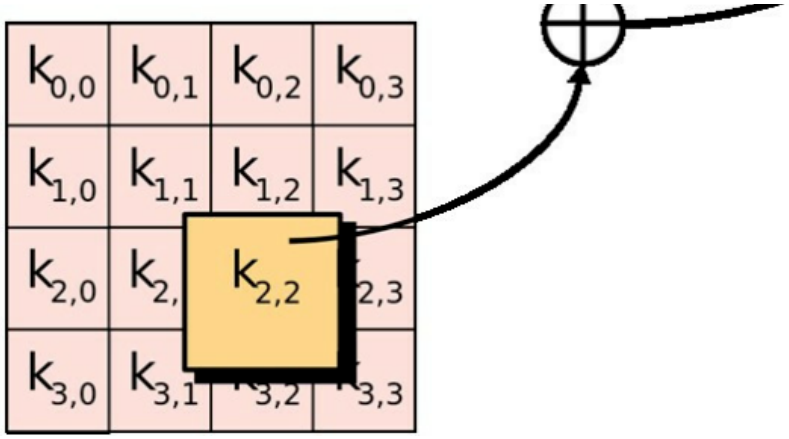
This is a 128-bit, 192-bit, or 256-bit variable created by an algorithm.

Cipher

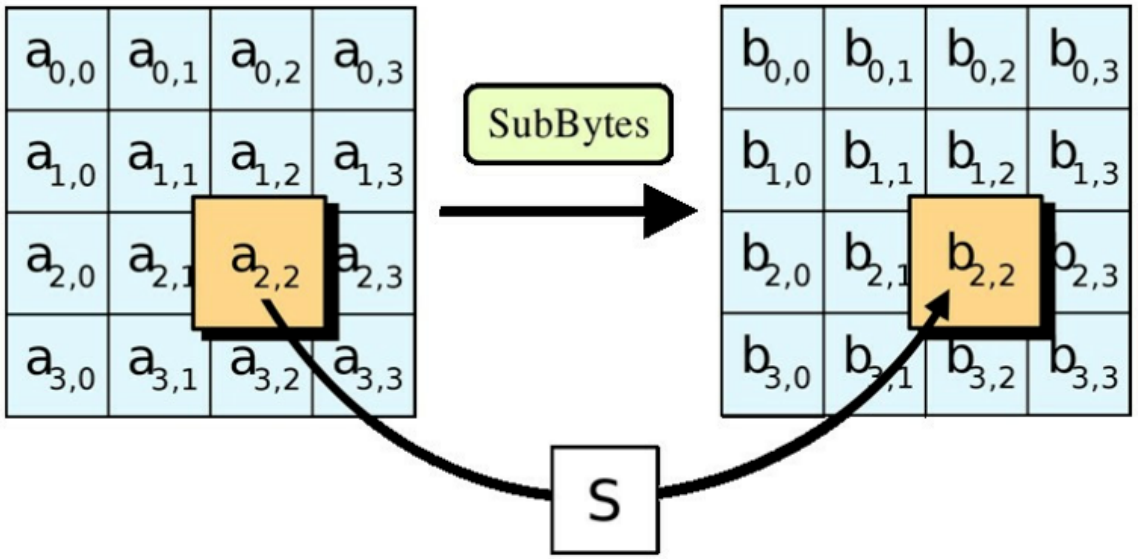
The actual AES cipher then performs a series of mathematic transformations using the plaintext and the secret key as a starting point. In order, these are:

1. Key expansion. This uses the original secret key to derive a series of new "round keys" using the Rijndael's key schedule algorithm.
2. Mixing. Each round key is combined with the plaintext using the additive XOR algorithm.

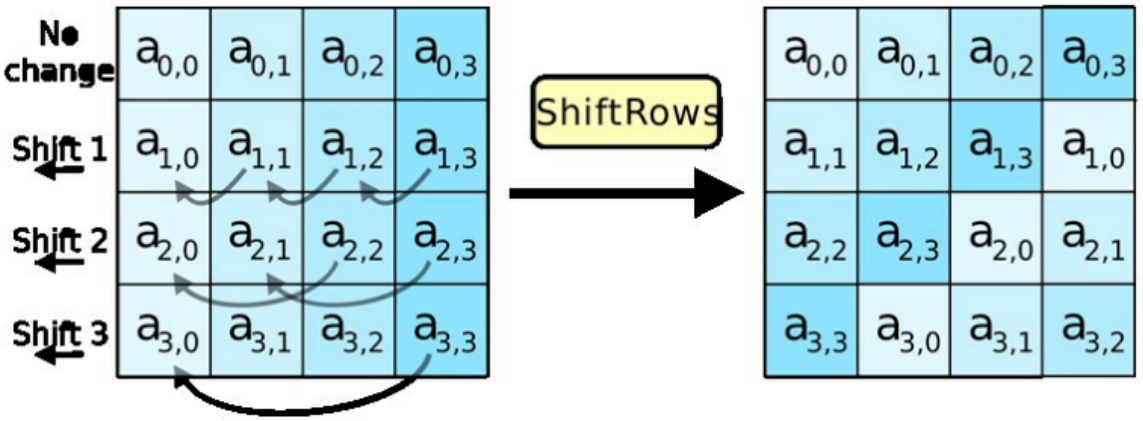




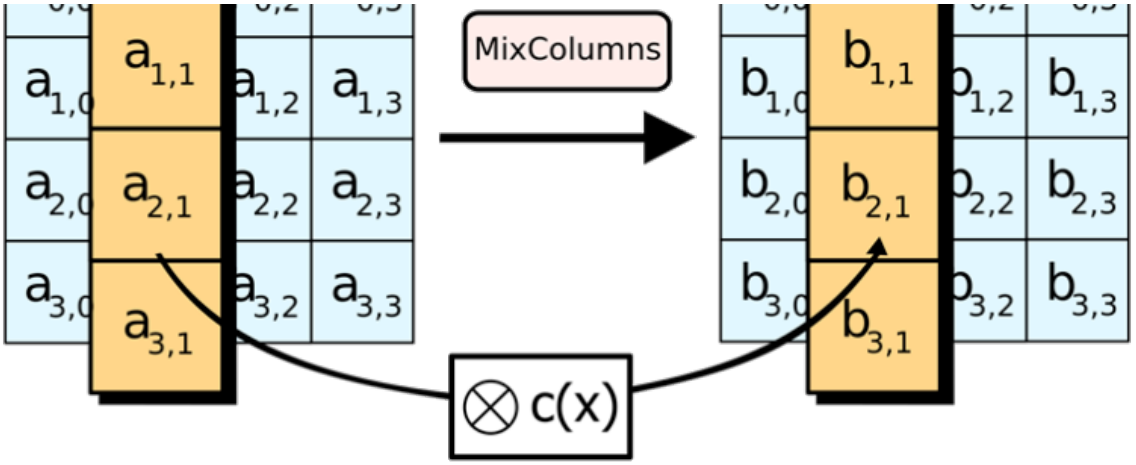
3. Substitution of the resultant data using a substitution table. This step is very similar in principle (if much more complex in practice) to the substitution ciphers you created as a kid.



4. Shift rows. In which every byte in the 4 x 4 column of sixteen bytes that makes up a 128-bit block is shifted to the right.



5. Mix columns. A further algorithm is applied to each column.



Rise and repeat. The process is repeated a number of times, with each repeat known as a round. Each round is re-encrypted using one of the round keys generated during key expansion (step 1).

The number of rounds performed depends on the key length used. AES-128 uses ten rounds, AES-192 uses twelve rounds, and AES-256 uses fourteen rounds.

Each added round reduces the chance of a shortcut attack of the kind that was used to attack AES-128 back 2011. As already noted as a consequence of this attack an additional four rounds were added to AES-128 in order to improve its safety margins.

Cipher text

This is the encrypted output from the cipher after it has passed through the specified number of rounds.

How to Decrypt AES encryption

Decrypting AES is simple – just reverse all the above steps, starting with the inverse round key. Of course, you need to have the original secret key in order to reverse the process using each inverse round key.

Does encrypting a file make it larger?

Yes. Usually. AES uses a fixed block size of 16-bytes. If a file is not a multiple of a block size, then AES uses padding to complete the block.

In theory, this does not necessarily mean an increase in the size of encrypted data (see ciphertext stealing), but simply adding data to pad out the block is usually much easier. Which increases the amount of data which is encrypted.

Anecdotal evidence suggests that files larger than 1 MB encrypted with AES tend to be around 35% larger than before encryption.

How important are key sizes in AES encryption?

The crudest way to measure the strength of a cipher is by the size of its key. The larger the key the more possible combinations there are.

AES is can be used with 126-bit, 192-bit, or 256-bit key sizes. The original Rijndael cipher was designed to accept additional key lengths, but these were not adopted into AES.

Brute force attacks

The more complex the algorithm, the harder the cipher is to crack using a brute force attack. This very primitive form attack is also known as an exhaustive key search. It basically involves trying every combination of numbers possible until the correct key is found.

As we are sure you know, computers perform all calculations using binary numbers: zeros and ones. And as we have seen, the complexity of a cipher depends on its key size in bits - the raw number of ones and zeros necessary to express its algorithm, where each zero or one is represented by a single bit.

This is known as the key length, and also represents the practical feasibility of successfully performing a brute force attack on any given cipher.

The number of combinations possible (and therefore the difficulty of brute force them) increases exponentially with key size. For AES:

Key Size	Possible combinations
1-bit	2
2-bit	4
8-bit	256
16-bit	65536
64-bit	4.2 x 10^9
128-bit	3.4 x 10^38
192-bit	6.2 x 10^57
256-bit	1.1 x 10^77

As we have already discussed, it would take the fastest supercomputer in the world longer than the age of the universe to crack even an AES-128 key by force!

Encryption rounds

As we have also discussed, the longer the key used by AES, the more it encryption rounds it goes through. This is primarily to prevent shortcut attacks which can reduce the computational complexity of ciphers, and which therefore make it easier to brute force the cipher.

As renounced cryptographer Bruce Schneier said of the 2011 shortcut attack on AES-128,

"Cryptography is all about safety margins. If you can break *n* round of a cipher, you design it with 2*n* or 3*n* rounds."

He did recommend introducing more rounds for each key size to AES, but NIST deems the current levels sufficient.

So why use more than AES-128?

All of which begs the question: if it would take longer than the age of the universe to crack even AES-128, why bother using AES-192 or AES-256? As Schneier noted:

"I suggest that people don't use AES-256. AES-128 provides more than enough security margin for the foreseeable future. But if you're already using AES-256, there's no reason to change."

Indeed, Schneier has argued in the past that AE-128 is, in fact, more secure that AES, because it has a stronger key schedule than AES-256.

So why is AES-256 held up as the gold standard of symmetric key encryption?

Safety margins

The 2011 shortcut attack demonstrates that no matter how secure experts think a cryptograph algorithm to be, inventive people will always find ways that nobody ever thought of to weaken them.

As with the number of rounds used, a larger key size provides a higher safety margin against being cracked.

Bling

The effect of marketing should not be ignored when considering the ubiquitousness of AES-256 encryption. The simple fact that AES-256 is widely regarded as the most secure symmetric encryption cipher in the world makes it the number one choice for many.

I mean, if AES-128 is good, then it only stands to reason that AES-256 must be better, right?

The fact the US government uses AES-256 to secure its most sensitive data only adds to its "bling" value, and allows VPN companies and the like to claim they use "military grade" encryption.

Given that this "bling perception" is (largely) accurate, there is little harm in the popularity of AES-256 (although see notes on NIST below).

AES and OpenVPN

VPN users, in particular, however, should be careful. Most VPN services use AES-256 to secure data transmitted by the OpenVPN protocol, but this is one of the various mechanisms used by OpenVPN to keep data secure.

A TLS connection secures transfer of the encryption keys used by AES to secure data when using OpenVPN. So if the OpenVPN TLS (control channel) settings are weak, then the data can become compromised despite being encrypted using AES-256. Please see our Ultimate Guide to VPN Encryption for more details.

AES-CBC vs AES-GCM

Until recently the only AES cipher that you were likely to encounter in the VPN world was AES-CBC (Cipher Block Chaining). This refers to the block cipher mode, a complex subject that is not really worth going into here.

Although CBC may theoretically have some vulnerabilities, the consensus is that CBC is secure. CBC is, indeed, recommended in the OpenVPN manual.

OpenVPN now also supports AES-GCM (Galios/Counter Mode).

GCM provides authentication, removing the need for an HMAC SHA hashing function. It is also slightly faster than CBC because it uses hardware acceleration (by threading to multiple processor cores).

AES-CBC remains the most common mode in general use, but AES-GCM is increasing in popularity. Given the advantages of GCM, this trend is only likely to continue. From a cryptographic perspective, though, both AES-CBC and AES-GCM are highly secure.

NIST

AES is a NIST-certified standard. This is a body that by its own admission works closely with the NSA in the development of its ciphers.

Given what we now know of the NSA's systematic efforts to weaken or build backdoors into international encryption standards, there is every reason to question the integrity of NIST algorithms. NIST, of course, strongly refutes such allegations:

"*NIST would not deliberately weaken a cryptographic standard.*"

It has also invited public participation in a number of upcoming proposed encryption standards, in a move designed to bolster public confidence.

The New York Times, however, accused the NSA of circumventing NIST-approved encryption standards by either introducing undetectable backdoors or subverting the public development process to weaken the algorithms.

This distrust was further bolstered when RSA Security (a division of EMC) privately told customers to stop using an encryption algorithm that reportedly contains a flaw engineered

by the NSA. This algorithm had also been endorsed by NIST.

Furthermore, Dual_EC_DRBG (Dual Elliptic Curve Deterministic Random Bit Generator) is an encryption standard engineered by NIST. It has been known to be insecure for years.

In 2006 the Eindhoven University of Technology in the Netherlands noted that an attack against it was easy enough to launch on "an ordinary PC." Microsoft engineers also flagged up a suspected backdoor in the algorithm.

Despite these concerns, where NIST leads, the industry follows. This is in large part due to the fact that compliance with NIST standards is a prerequisite to obtaining US government contracts (FIPS).

NIST-certified cryptographic standards such as AES are pretty much ubiquitous worldwide, throughout all areas of industry and business that rely on privacy. This makes the whole situation rather chilling.

Perhaps precisely because so much relies on these standards, cryptography experts have been unwilling to face up to the problem.

Image credit: xkcd.com/538.