No Weak Links: A White Paper on the Lectrosonics D-Series

Version 1.1, February 12, 2014

In this age of wireless internet and online banking, people generally understand the need for secure data transmission. But other than looking for that reassuring padlock icon or selecting a "strong" password, most of us have little exposure to the technical underpinnings of data security. We place our trust in reputable companies, products and services, and leave the details to the experts.

Choosing an encrypted wireless microphone isn't like choosing an internet router, however. It can be much more difficult to know which are the reputable companies, products and services, and who are the experts. Lectrosonics is an established industry leader with a reputation for excellent product quality, reliability, and customer service. If this is all you need to know, then we are deeply honored, and grateful for your patronage. If you would like more information about the data security aspect of digital wireless microphone systems, we are very proud to offer you this overview.

No Weak Links

What does it take to keep audio secrets? A lot of things. A good method of encryption and large key size are only part of the complete picture. Other important considerations, such as key management, entropy sources and the *mode* of encryption, are equally important, yet rarely disclosed. Our design approach has been defined by the wise words, "A chain is only as strong as its weakest link," aggressively applied. If you are considering an encrypted wireless purchase, here are some of the "links" you will want to make sure are strong.

Encryption Algorithm and Key Size

The algorithm and key size are usually the most visible and well understood features of any encryption system. We have chosen the extremely well established AES-256 (Advanced Encryption Standard with 256-bit key).[i] This algorithm, first published in 1998, was approved as a U.S. government standard in 2001 [FIPS 197].[ii] It has been used by hundreds of millions of people worldwide, analyzed and attacked continually by commercial experts and academic institutions, and as of this publication date (early 2014), is not known to have any substantial flaws.[iii] Remarkably, this remains true even in the wake of the infamous "leaks" to the newsmedia in 2013 revealing vulnerabilities in other previously trusted systems.[iv]

A 256-bit key length means that the number of possible keys is 2 to the 256 power, or approximately 116 thousand trillion, trillion, trillion, trillion, trillion, *trillion*. That's a one followed by 77 zeros! With a good cryptographic system such as AES, there is no such thing as an incorrect key that is "close" to the correct one. That means guessing is useless, for all practical purposes. Even supposing that some

supercomputer could test a trillion keys every nanosecond, it would still take more than 3 trillion, trillion, trillion, *trillion* years to try them all.

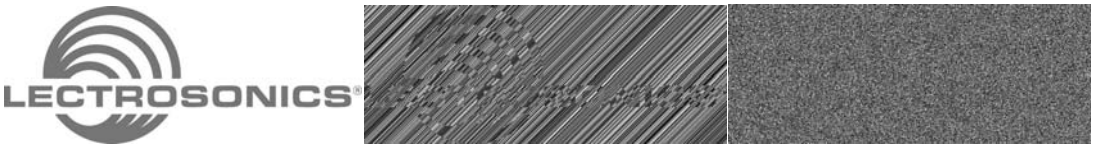
Does the use of AES-256 make your data safe? Not necessarily. It is merely one extremely strong link in a chain. Read on to learn of other potential "weak links" which users of our D-Series encrypted wireless systems need never encounter.

Encryption Mode

If you wanted to use a padlock to protect a diamond ring, you might lock it in a storage locker (excellent security), lock it in a small box (not as good), or simply secure the padlock directly to the ring (laughably poor security). Just as the same lock can protect something well or poorly, depending on how it is used, encryption algorithms can be used in various ways, and some work much better than others. Below are some descriptions of commonly used modes.[v]

ECB (Electronic Codebook): Commonly used but not secure

This lofty-sounding name describes the simplest approach: each piece of data is encrypted separately using the same key. The trouble with this unsophisticated technique is that it violates a fundamental tenet of cryptography: *Never encrypt anything the same way twice*. Violating this principle makes possible something called a "differential attack". While that might sound like merely a theoretical threat, here is a striking graphical demonstration of the potential folly of ECB mode.[vi]



From left to right: unencrypted, ECB mode encrypted, CTR mode encrypted

ECB Advantages: simplicity
ECB Pitfalls: highly vulnerable -- not recommended

CBC (Cipher-Block Chaining): Good security but has other issues

This technique avoids the dangers of ECB mode by using the result of each encryption to perturb the next. The result is much better security, but there are still pitfalls, especially for wireless applications. Special care must be taken to avoid key reuse if the transmitter is powered off an on again. Also, because this mode encrypts a "block" at a time (typically a millisecond or two of audio), it adds latency (time delay), and changes tiny signal defects (single bit errors) into long noise bursts.

CBC Advantages: good security
CBC Pitfalls: possible key reuse at powerup, adds 2 to 4 ms latency, can exaggerate channel noise

CTR (Counter): Highly recommended

The Lectrosonics D-Series uses CTR mode. CTR mode cleverly turns the encryption algorithm on its head, encrypting a counter (which never repeats), and then using that encrypted counter value as a "key" to encrypt the audio. This counterintuitive method (pun intended) offers several important advantages. One is that we are able to ensure that no counter value is reused over the entire life of the

equipment, thus slamming the door on differential attacks. Another is that no latency is added by the encryption system, because counter values can be encrypted in advance. A third advantage is that single bit errors remain single bit errors, so a noisy channel sounds no worse than it absolutely has to. About the only disadvantage of CTR mode is that it was formerly believed to be less secure, because it was incompatible with many old-fashioned encryption algorithms. Modern encryption algorithms such as AES are fully compatible with CTR mode. [vii]

CTR Advantages: key reuse easily avoided, minimum latency, no exaggeration of channel noise
CTR Pitfalls: system must prevent counter value reuse, algorithm must be compatible with CTR mode

Key Management

In an encrypted wireless system, the transmitter and receiver must agree on the encryption key. The way this key is communicated and stored is another of those potential "weak links," and thus influences the ultimate security of the system. For example, if the key is communicated using IR (infrared light) signals, theoretically a well-placed IR detector (available on learning remotes and some smart phones) can intercept it. For this reason, the Lectrosonics D-Series uses a cable for key exchange. Technology does exist that can intercept the cable signal as well, but that is far more range-limited, specialized and expensive, placing it out of reach of the vast majority of would-be attackers.[viii]

If the key is stored in non-volatile memory, so it may be used when the system is powered back on, it may be possible for someone with physical access to the hardware, even much later, to retrieve the key and decrypt one or more transmissions. The Lectrosonics D-Series leaves this choice up to the user or installer: the key may be stored, or a new key may be generated for each session which is never stored.

Finally, if a given key may be exchanged more than once, or if any internal protocols exist wherein the hardware may be asked to reveal the key, these are potential exploits. By design, the Lectrosonics D-Series has no such vulnerabilities.

Entropy Source

One more potential "weak link" has to do with the method for generating a key. A good encryption algorithm cannot offer optimum security unless all possible keys are equally likely to be used. Any lack of "randomness" can allow an attacker to take massive shortcuts, trying the most likely keys first.

To illustrate this point, let's consider a truly woeful method of key generation: For each bit in the key, roll two six-sided dice. If the dice come up with two sixes, the bit shall be a "1", otherwise the bit shall be a "0". In theory, this method can produce any possible key; however the overwhelming majority will consist of almost all zeros. Even inexpensive modern hardware would have a 50% chance of guessing the correct key in a few minutes. That is one very weak link! All real key generation methods are better than this one, but few are truly random, and even a small defect in this process goes a long way to make the attacker's job easier.

True randomness (or, more technically, "entropy") is quite a commodity in our orderly universe. Events may appear random, yet exhibit patterns which make them unsuitable for security purposes. Some commonly used poor sources of entropy include human attempts at randomness, pseudo-random

sequence generators, and certain types of noise from sensors and converters. Some good sources of entropy include diode noise (upon which principle many "True Random Number Generator" circuits are based), radioactive decay, and metastable electronic circuits. Even these "good" sources must be carefully evaluated and adjusted to ensure desirable statistical properties over all operating conditions. To sum up, it is hard to avoid compromising a security system with an imperfect entropy source.[ix]

The Lectrosonics D-Series takes no chances with this vital link in your data security chain. We've left the challenge of harvesting true randomness to the experts. Every Lectrosonics D-Series system comes with a specialized third-party chip installed which has been certified to meet government standards for randomness, specifically for the purpose of data security [FIPS 140-2].[x]

Conclusion

Lectrosonics has built some of the most dependable wireless radio links in the industry, and our approach to data security is every bit as uncompromising. Our goal is for your audio signal and data security chains to have no weak links.

Notes

[i] http://en.wikipedia.org/wiki/Advanced_Encryption_Standard
[ii] http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
[iii] http://blog.agilebits.com/2011/08/18/aes-encryption-isnt-cracked/
[iv] http://www.pcpro.co.uk/blogs/2013/09/06/has-the-nsa-really-broken-strong-encryption/
[v] http://en.wikipedia.org/wiki/Block_cipher_mode_of_operation
[vi] http://bobnalice.wordpress.com/2009/01/28/friends-don%E2%80%99t-let-friends-use-ecb-mode-encryption/
[vii] http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/proposedmodes/ctr/ctr-spec.pdf
[viii] http://en.wikipedia.org/wiki/Tempest_(codename)
[ix] https://www.ietf.org/rfc/rfc4086.txt
[x] http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf