Unambiguous Encapsulation

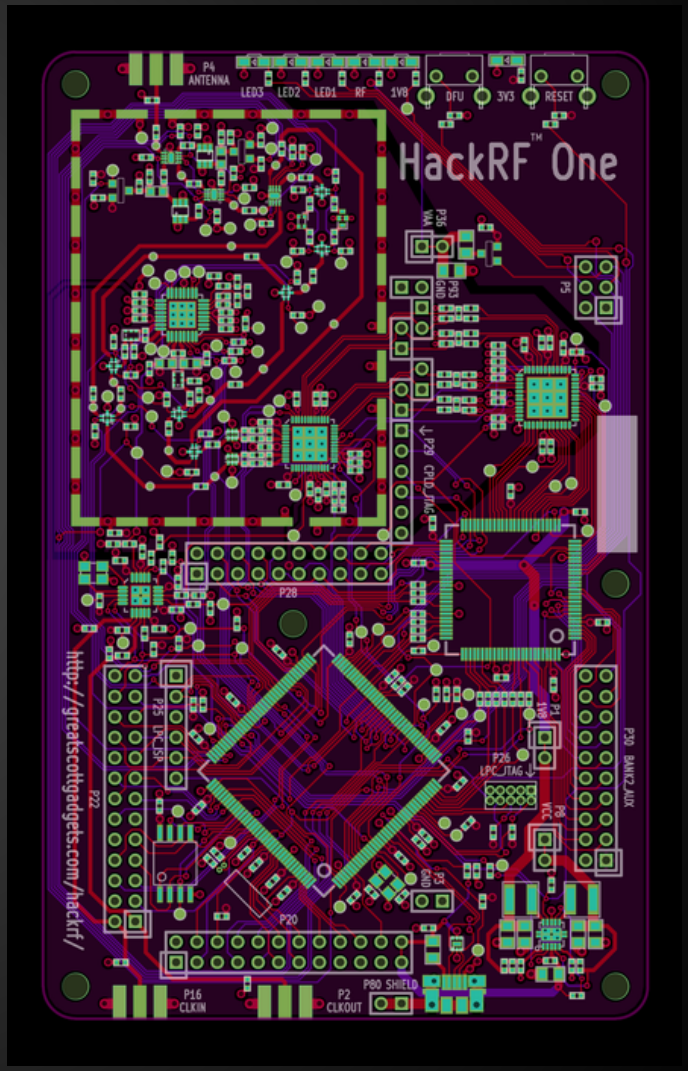
Separating Data and Signaling
LangSec workshop 2015

Michael Ossmann

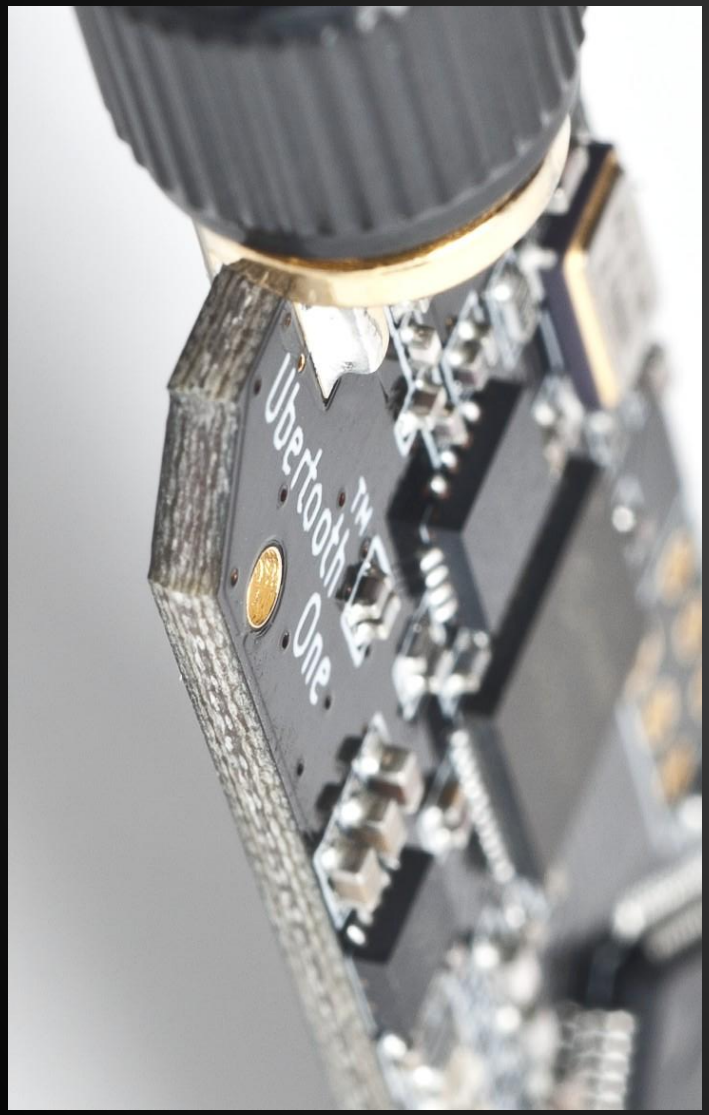
Primary on Unambiguous Encapsulation

Creator of multiple OSHW projects, Ubertooth, HackRF, Daisho, YARD Stick One

Founder of Great Scott Gadgets



Dominic Spill



Code for Unambiguous Encapsulation

Dev on Ubertooth, BTBB, gr-bluetooth, Daisho, USBProxy

Other projects include BeagleDancer, PS/2 tap and fcc.io

Disclaimer

The views expressed are the views of the authors and do not reflect the official policy or position of the Department of Defense or the United States Government.

Outline

The Problem

Unambiguous Encapsulation

Error Control Codes

Finding Interesting Error Control Codes

Background

LANGSEC

Packets in Packets

The Problem - Packets in Packets

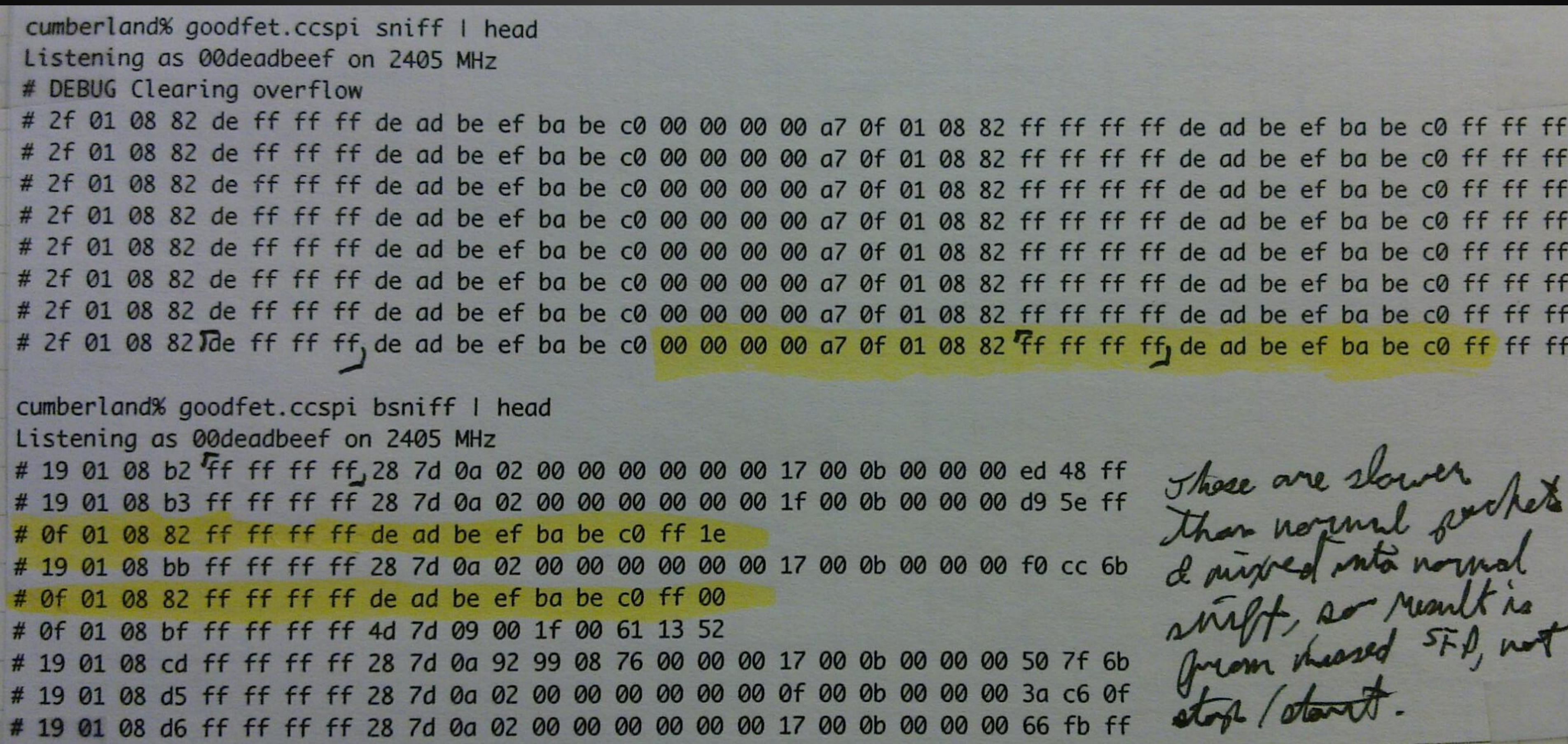
Interference or glitch obscures packet header

Second packet in payload

Receiver detects second packet

Zigbee / Ethernet susceptible

The Problem - Packets in Packets

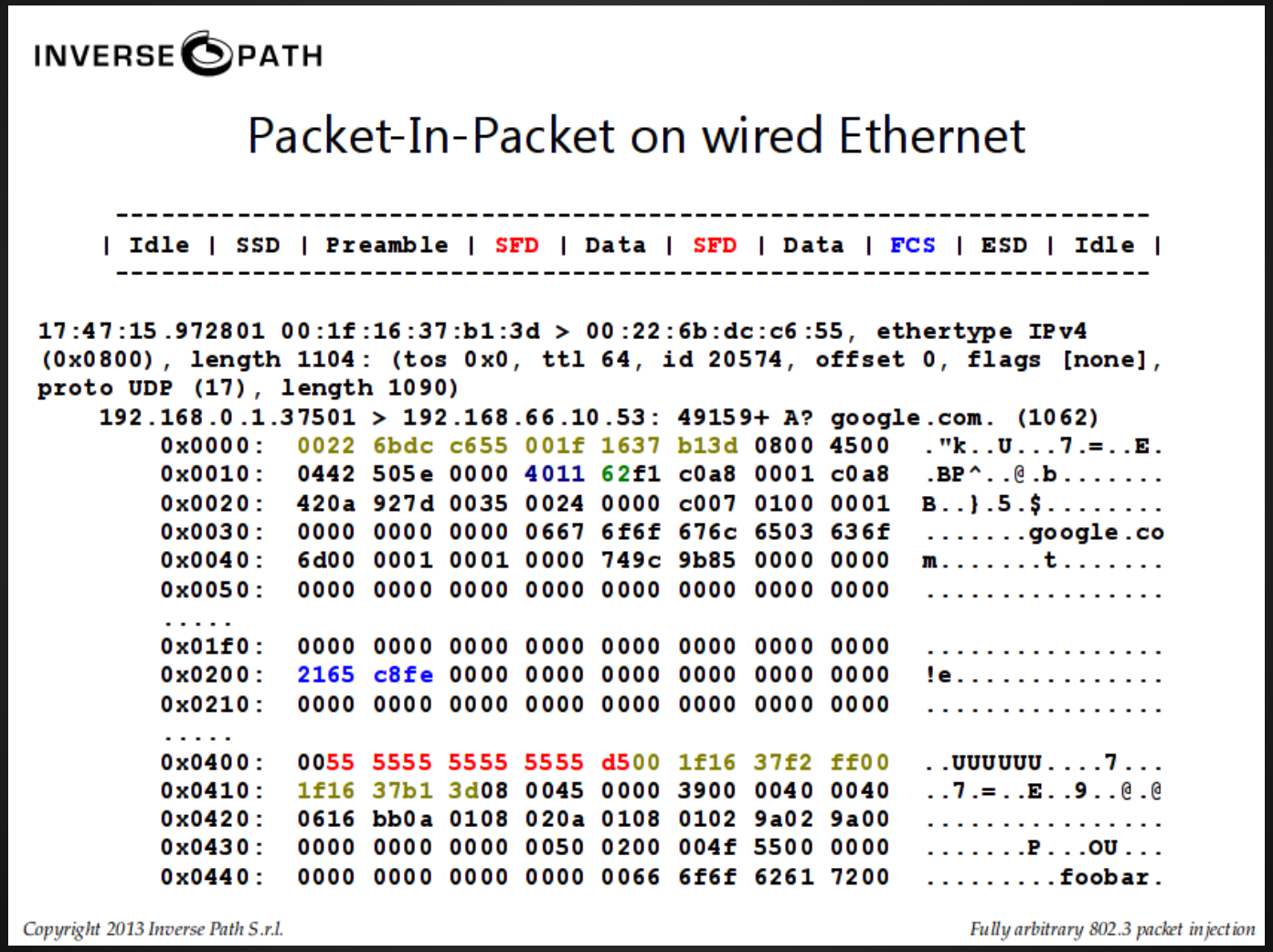


```
cumberland% goodfet.ccspi sniff | head
Listening as 00deadbeef on 2405 MHz
# DEBUG Clearing overflow
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff
# 2f 01 08 82 de ff ff ff de ad be ef ba be c0 00 00 00 00 a7 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff ff ff

cumberland% goodfet.ccspi bsniff | head
Listening as 00deadbeef on 2405 MHz
# 19 01 08 b2 ff ff ff ff 28 7d 0a 02 00 00 00 00 00 00 17 00 0b 00 00 00 ed 48 ff
# 19 01 08 b3 ff ff ff ff 28 7d 0a 02 00 00 00 00 00 00 1f 00 0b 00 00 00 d9 5e ff
# 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff 1e
# 19 01 08 bb ff ff ff ff 28 7d 0a 02 00 00 00 00 00 00 17 00 0b 00 00 00 f0 cc 6b
# 0f 01 08 82 ff ff ff ff de ad be ef ba be c0 ff 00
# 0f 01 08 bf ff ff ff ff 4d 7d 09 00 1f 00 61 13 52
# 19 01 08 cd ff ff ff ff 28 7d 0a 92 99 08 76 00 00 00 17 00 0b 00 00 00 50 7f 6b
# 19 01 08 d5 ff ff ff ff 28 7d 0a 02 00 00 00 00 00 00 0f 00 0b 00 00 00 3a c6 0f
# 19 01 08 d6 ff ff ff ff 28 7d 0a 02 00 00 00 00 00 00 17 00 0b 00 00 00 66 fb ff
```


Credit: Travis Goodspeed

Ethernet Too!



Credit: Andrea Barisani and Daniele Bianco

The Problem - Buffer Overflow

User supplied data written to buffer

Overwrite data on stack

CPU executes data as instructions

Ambiguous Encapsulation

Given a piece of data without context, it is not possible to determine if it is meta-data or encapsulated data

Unambiguous Encapsulation

Given a piece of data without context, it is possible to determine if it is meta-data or encapsulated data

If you haven’t found the analog medium beneath a particular bit or byte, keep digging

Error Control Codes

Error control codes are used at the boundary between analog and digital

Can we find error control codes that provide useful encapsulation properties?

Error Control Codes

Encapsulate data in codewords

Binary Linear Block Codes encode *k* data bits in *n* bit codewords with a minimum Hamming distance *d*

Often designated by [*n*,*k*] or [*n*,*k*,*d*]

[7,4,3] Hamming Code

0000000	0101010	1000011	1101001
1011010	0110011	1110000	0011001
1001100	1100110	0001111	0100101
0111100	0010110	1111111	1010101

Each codeword is 7 bits long, $n = 7$
There are 2^4 codewords, $k = 4$
At least 3 bits differ between any two codewords, $d = 3$

[7,4,3] Hamming Code

codeword length = 7

number of codewords = 2^4

minimum Hamming distance = 3

One bit flipped: error corrected

Two bits flipped: error detected

Three bits flipped: undetected error

Implementation

[7,4,3] Hamming encoder:

look-up table: 16 * 7 bits

[7,4,3] Hamming decoder:

look-up table: 128 * 4 bits

Much of the complexity of coding theory is related to clever decoding methods, but a look-up table works for shorter (small *n*) codes

Brute Force Coding

Decoding by look-up table is sort of a brute force approach

We can also take a brute force approach to the discovery of new codes

A [5,3,2] Code

00000		01110	
	00011		10110
00101		11010	
	01001		11100

Hamming Distance = 2

Isolation

A code can be thought of as a pair of complementary sub-codes.

A [5,3,2,3] Isolated Complementary Binary Block Code (ICBBC)

codeword length = 5

number of codewords = 2^3

minimum Hamming distance = 2

minimum isolation = 3

One bit flipped: error detected

Two bits flipped: undetectable error, isolated

Three bits flipped: isolation broken

Searching for codes

C program to brute force search for codes

Depth First Search recursive algorithm

Other search methods

FPGA

- Verilog implementation shows promise
- Recursion difficult unless we know max depth of recursion at compile time

Z3 Python

- Implementation of icbbc search exist
- Fast non-exhaustive search

ICBBC Search

Search space

- $2 \leq n \leq 8$
- $1 \leq$ Hamming distance $\leq n$
- Hamming distance $\leq$ isolation $\leq n$

Some larger codes sought

- [15,7,9] code produce 111GB of output

ICBBC Search Results

Results

19,189,776 codes found

Symmetric codes

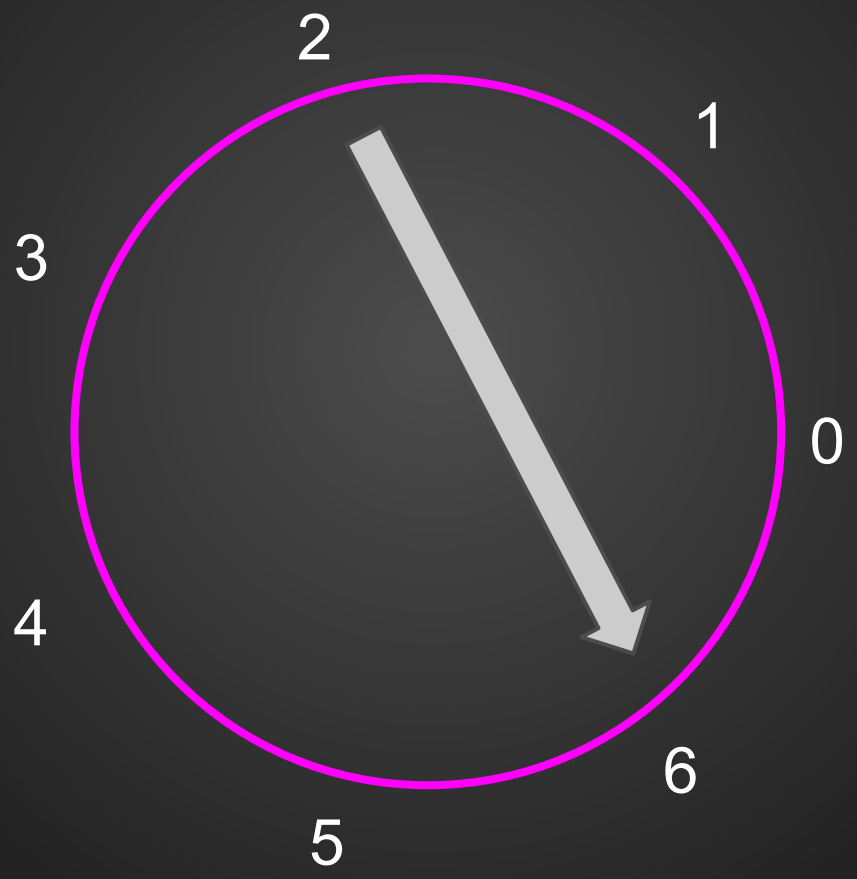
[8,3,5] / [8,4,5] - similar subcode sizes

Asymmetric codes

[8,2,5] - subcodes of 2 / 44 codewords

Isolated Complementary Non-Binary Block Codes (ICNBC)

a 7-PSK example



Lee Distance from 2 to 6 is 3

Lee Distance from (2,1,3) to (6,6,6) is 3+2+3=8

ICNBC Examples

2 symbol codewords, minimum Lee distance of 2, isolation of 5:

[(0, 0), (1, 1)] [(3, 4), (4, 3), (4, 5), (5, 4)]

3 symbol codewords, minimum Lee distance of 5, isolation of 7:

[(0, 0, 0), (4, 6, 6)] [(3, 3, 2), (6, 4, 3)]

ICNBC Search

Largest search space of the project

- 11,000 sets of parameters
- $2 \leq n \leq 9$
- $1 \leq$ Lee distance $\leq 2n$
- $1 \leq$ isolation $\leq 2n$

ICNBC Search Results

Results

- 20GB of successful output
- Some processes constrained by resources

Very few symmetric codes

Asymmetric codes

- [5,1,8] - subcodes of 2 / 10,264 codewords
- [5,2,15] - subcodes of 2 / 4 codewords

Large Complementary Binary Block Codes (LCBBC)

Sometimes the largest binary block code for a given codeword length and Hamming distance is not a power of two.

Example: 8 bit codewords, minimum Hamming distance of 3, 20 codewords:

[0, 7, 25, 30, 42, 53, 75, 84, 97, 108, 114, 127, 140, 147, 166, 169, 176, 194, 197, 216]

LCBBC Search

Search space

- $2 \leq n \leq 15$
- $2 \leq$ Hamming distance $\leq n$

LCBBC Search Results

Smallest set of search results

- ~1MB
- Smaller search space
- Only searching for longest code

Code Selection

Error control codes are typically selected based on:

- code rate (*k*/*n*)
- complexity of decoder
- probability of undetectable error
- probability of uncorrectable error

We suggest an addition to this list:

- probability of encapsulation breakage

Future Work

Implementations
- gr-802.15.4
- Ethernet using Daisho
- Harvard architecture / NX replacement

Additional code classes

Investigate the nature of noise

Unambiguous Encapsulation

Any time you encapsulate data within other data, consider unambiguous encapsulation

Thank You

LANGSEC community
DARPA Cyber Fast Track
Sergey Bratus
David Hulton
Mike Kershaw
Tariq Bashir Ahmad

Questions?

http://github.com/mossmann/unambiguous-encapsulation

Twitter:

@michaelossmann

@dominicgs