Computer Forensics

Kickstart Training

Michael Hamm - CIRCL

TLP:CLEAR

January, 2025

2.1 Data in a binary system

- BIT $\rightarrow$ Binary digit
- Data stored in binary form

  `x Bits --> 01010000011010010110111001100111 --> y Bits`

  Bit x + 2 = 1

  Bit x + 3 = 0

  $\rightarrow$ What information is stored within this data?

- *"..... information is data arranged in a meaningful way for some perceived purpose ....."* $\rightarrow$ Interpretative rules

- Grouping, addressing and interpreting

```
-->   01010000   01101001   01101110   01100111   -->
      --------   --------   --------   --------
-->   Byte 117   Byte 118   Byte 119   Byte 120   -->
```

2.1 Data in a binary system

- Grouping examples:
  - Nibble: `0101 0000 0110 1001 0110 1110 0110 0111`
  - Byte: `01010000 01101001 01101110 01100111`
  - Word: `0101000001101001 0110111001100111`
  - Double Word: `01010000011010010110111001100111`

- Interpreting:
  - Integer: (Signed, Unsigned)
  - Endian: (Big, Little)
  - Floating Point
  - Binary Coded Decimal, Packed BCD
  - Encoding: (ASCII, ISO8859, Unicode 16L, 16B, 32L, 32B)
  - Binary: (ELF, MZ, PE, GIF, JPEG, ZIP, PDF, OLE, ...)
  - ...

2.2 Number Systems

- Decimal:

```
2145
||||_   5 * 10^0 =      5
|||_--   4 * 10^1 =     40
||_---   1 * 10^2 =    100
|_----   2 * 10^3 =  2,000
                    -------
                    2,145
```

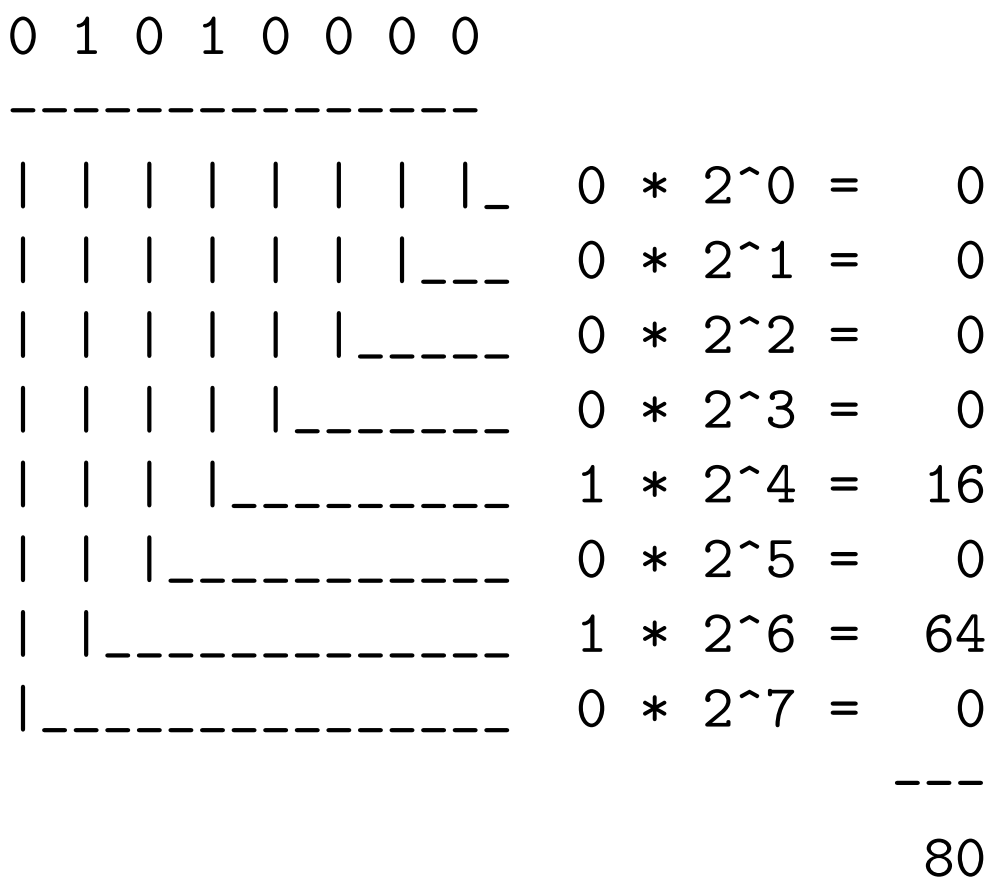
- Binary:

```
1111
||||_   1 *  2^0 =      1
|||_--   1 *  2^1 =      2
||_---   1 *  2^2 =      4
|_----   1 *  2^3 =      8
                    -------
                         15 = 1111
```

- Hexadecimal:

```
2A9F
||||_  15 * 16^0 =     15
|||_--  09 * 16^1 =    144
||_---  10 * 16^2 =  2,560
|_----  02 * 16^3 =  8,192
                    -------
                    10,911 = 0x2A9F
```

2.3 Interpreting binary data: Integer



2.3 Interpreting binary data: Signed Integer

```
1 0 1 1 1 1 1 1
---------------
| | | | | | | |

0 1 0 0 0 0 0 0
0 1 0 0 0 0 0 1

  |           |

 64           1
 -------------
             -65
```

Two's complement:

1. Invert all single bits
2. Add the value 1

3. Convert to Decimal

2.4 Exercise: Signed Integer Bytes

```
1 1 0 1 1 1 0 0
---------------
| | | | | | | |


    |     |

    ?     ?
---------------
              -??
```

Two's complement:

1. Invert all single bits
2. Add the value 1

3. Convert to Decimal

2.4 Exercise: Signed Integer Bytes

```
1 1 0 1 1 1 0 0
---------------
| | | | | | | |

0 0 1 0 0 0 1 1
0 0 1 0 0 1 0 0

    |     |

   32     4

--------------
             -36
```

Two's complement:

1. Invert all single bits
2. Add the value 1

3. Convert to Decimal

2.4 Exercise: Challenge on 1 byte signed Integer

- Find biggest possible positive number

  —>

- Find smalest possible positive number

  —>

- Find biggest possible negative number

  _____

  _____ —>

- Find smalest possible negative number

  _____

  _____ —>

2.4 Exercise: Challenge on 1 byte signed Integer

- Find biggest possible positive number

  0111 1111 —> 127

- Find smalest possible positive number

  0000 0000 —> 0

- Find biggest possible negative number

  _____

  _____ —>

- Find smalest possible negative number

  _____

  _____ —>

2.4 Exercise: Challenge on 1 byte signed Integer

- Find biggest possible positive number

  0111 1111 —> 127

- Find smalest possible positive number

  0000 0000 —> 0

- Find biggest possible negative number

  1111 1111

  0000 0000
  0000 0001 —> −1

- Find smalest possible negative number

  1000 0000

  0111 1111
  1000 0000 —> −128

2.5 From Bin to Hex

Example:

0001 1000	0101 0101	0000 1111	1010 0110
1 8	5 5	0 F	A 6

Exercise:

1001 0110	1010 0101	0000 1111	1100 0011

2.5 From Bin to Hex

Exercise:

```
1001 0110    1010 0101    0000 1111    1100 0011
---------    ---------    ---------    ---------
```

Results:

```
1001 0110    1010 0101    0000 1111    1100 0011
---------    ---------    ---------    ---------
   9 6          A 5          0 F          C 3
```

2.6 Big Endian and Little Endian

Multibyte words:

Example: 256 in Big Endian representation:

2^:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Address:	10.000								10.001							

Multibyte words:

Example: 256 in Little Endian representation:

2^:	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8
Data:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Address:	10.000								10.001							

2.6 Exercise: Little Endian

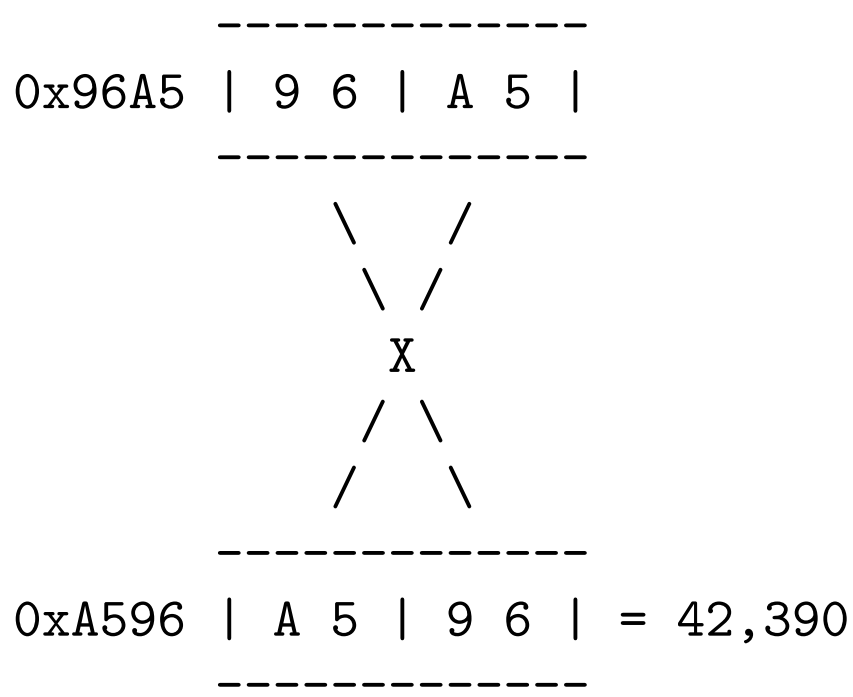
Read and interpret this little endian 2 byte 'word'

```
          -------------
0x96A5 | 9 6 | A 5 |
          -------------
```

```
          -------------
0x        |     |     | =
          -------------
```

2.6 Exercise: Little Endian

Read and interpret this little endian 2 byte 'word'



2.6 Exercise: Little Endian

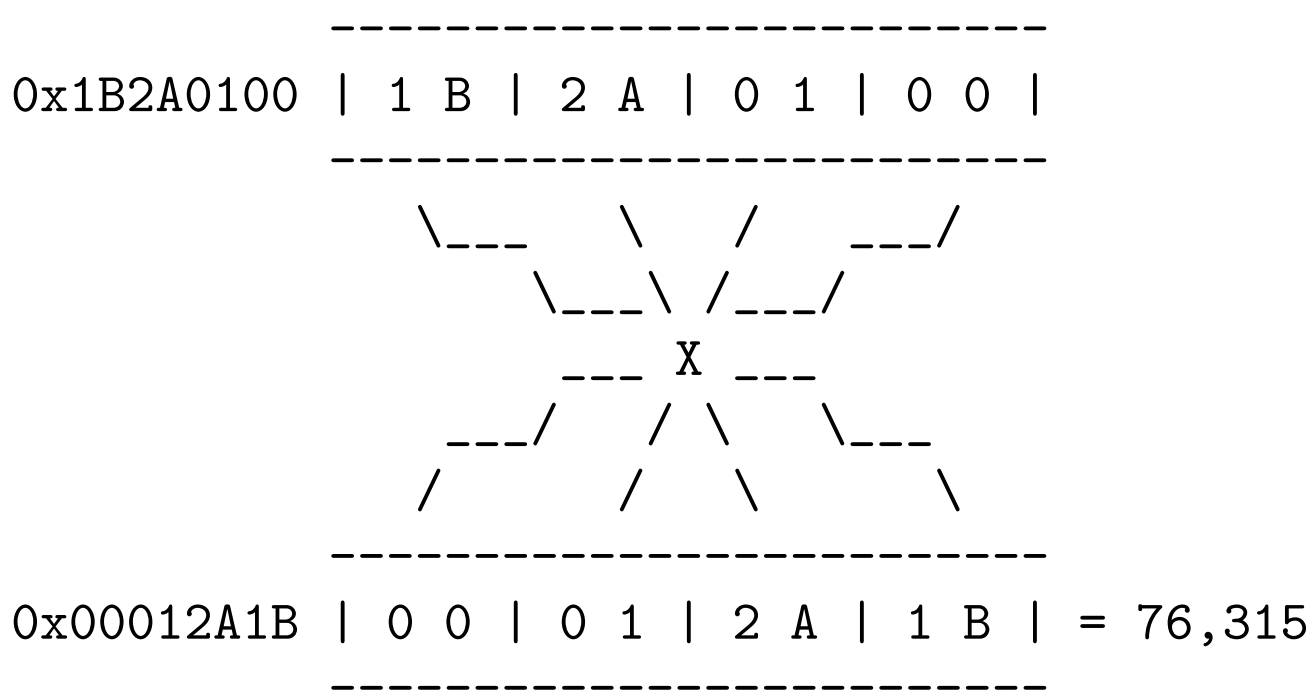
Read and interpret this little endian 'double word'

```
                ------------------------
0x1B2A0100 | 1 B | 2 A | 0 1 | 0 0 |
                ------------------------
```

```
                ------------------------
0x              |     |     |     |     | =
                ------------------------
```

2.6 Exercise: Little Endian

Read and interpret this little endian 'double word'



2.7 Example: Other interpretation of binary data

BCD / PBCD

```
   2        9        1       /     6  na    0  9
-------- -------- --------   /   -------- --------
00000010 00001001 00000001   /   01101010 00001001
```

ASCII

```
01110000  01101001  01101110  01100111
--------  --------  --------  --------
  0x70      0x65      0x6E      0x67
   112       105       110       103
    p         i         n         g
```

2.8 Data structures: Exercise

- Can you read this data?
- Can you extract information out of this data?
- Can you generate knowledge out of this data?

```
01000100010001100100010101000001000010000000111000000000011111111
10111010001100101011100110111010000101110011101000111100001110 1
00001000100100100001100101011011000110110001101111001000000101 0
11101101111011100100110110001100100001000100000110101000100010 0
01100100010101000001000001110000000100000001000000000011001000 11
00110011010010111001000101110001101100011010001010010010001010 1
01101001001010010101010110100101000010011110010110010001010111 0
11110000110110001100101011001110110111100111101000010101111111 1
11111111111111111111111
```

2.8 Data structures: Organizing data

0	8	16
44 46 45 41 08 0E 00 FF	74 65 73 74 2E 74 78 74	22 48 65 6C 6C 6F 20 57

24	32	40
6F 72 6C 64 22 0D 44 46	45 41 07 11 00 00 64 66	69 72 2E 36 34 52 45 5A

48	56	64
4A 55 69 42 79 64 57 78	6C 65 67 6F 3D 0A FF FF	FF FF

2.8 Data structures: Definition of the structure

0	8	16
44 46 45 41 08 0E 00 FF	74 65 73 74 2E 74 78 74	22 48 65 6C 6C 6F 20 57

24	32	40
6F 72 6C 64 22 0D 44 46	45 41 07 11 00 00 64 66	69 72 2E 36 34 52 45 5A

48	56	64
4A 55 69 42 79 64 57 78	6C 65 67 6F 3D 0A FF FF	FF FF

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (–1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.8 Data structures: Apply structure

```
 0                               8                               16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|
|               |   |       |   |

 24                              32                              40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|

 48                              56                              64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF|   |   |   |   |   |
```

```
Offset  Size  Description
     0     4  Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
     4     1  Lenght of file name (Integer)
     5     2  Lenght of data (Little Endian)
     7     1  Type of data (Signed Integer)    (–1 = ASCII; 0 = base64 encoded)
     8   ––   Variable file name (ASCII)
    9++  ––   Data (Binary)
_    EOF   4  EOF signature (Binary: FF FF FF FF)
```

2.8 Data structures: Read information

0	8	16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|		
| D F E A| | | |		

24	32	40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|		

48	56	64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF| | | | | | |		

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (–1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.8 Data structures: Read information

0	8	16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|		
| D F E A| 8| | | |		

24	32	40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|		

48	56	64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF| | | | | | |		

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (–1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.8 Data structures: Read information

```
 0                               8                               16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|
| D  F  E  A| 8|  14 |  |

 24                              32                              40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|

 48                              56                              64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF|  |  |  |  |  |  |
```

```
Offset  Size  Description
     0     4  Header signature (ASCII: DFEA - Digital Forensics EDU Archive)
     4     1  Lenght of file name (Integer)
     5     2  Lenght of data (Little Endian)
     7     1  Type of data (Signed Integer)    (-1 = ASCII; 0 = base64 encoded)
     8   --   Variable file name (ASCII)
    9++  --   Data (Binary)
_    EOF   4  EOF signature (Binary: FF FF FF FF)
```

2.8 Data structures: Read information

0	8	16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|		
| D F E A| 8| 14 |−1|		

24	32	40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|		

48	56	64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF| | | | | | |		

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (−1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.8 Data structures: Apply informtion

```
 0                               8                               16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|
| D  F  E  A| 8|  14 |-1|                                |

 24                              32                              40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|
                     |

 48                              56                              64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF|  |  |  |  |  |  |
```

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (–1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.8 Data structures: Interprete bytes

```
 0                           8                           16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|
| D  F  E  A| 8|  14 |-1| t  e  s  t  .  t  x  t |

 24                          32                          40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|
                     |

 48                          56                          64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF|  |  |  |  |  |  |
```

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (–1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.8 Data structures: Interprete bytes

```
 0                               8                               16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|
| D  F  E  A| 8|  14 |-1| t  e  s  t  .  t  x  t |"  H  e  l  l  o     W

 24                              32                              40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|
  o  r  l  d  " CR|

 48                              56                              64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF|  |  |  |  |  |  |
```

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (–1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.8 Data structures: Exercise: Your turn

```
 0                               8                               16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|
| D  F  E  A| 8|  14 |-1| t  e  s  t  .  t  x  t |" H  e  l  l  o     W

 24                              32                              40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|
  o  r  l  d  " CR|

 48                              56                              64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF|  |  |  |  |  |  |
```

```
Offset  Size  Description
     0     4  Header signature (ASCII: DFEA - Digital Forensics EDU Archive)
     4     1  Lenght of file name (Integer)
     5     2  Lenght of data (Little Endian)
     7     1  Type of data (Signed Integer)    (-1 = ASCII; 0 = base64 encoded)
     8   --   Variable file name (ASCII)
    9++  --   Data (Binary)
_    EOF   4  EOF signature (Binary: FF FF FF FF)
```

2.8 Data structures: Exercise: Solution

```
 0                               8                               16
|44|46|45|41|08|0E|00|FF|74|65|73|74|2E|74|78|74|22|48|65|6C|6C|6F|20|57|
| D  F  E  A| 8|  14 |-1| t  e  s  t  .  t  x  t |"  H  e  l  l  o     W

 24                              32                              40
|6F|72|6C|64|22|0D|44|46|45|41|07|11|00|00|64|66|69|72|2E|36|34|52|45|5A|
  o  r  l  d "CR| D  F  E  A| 7|  17 | 0| d  f  i  r  .  6  4| R  E  Z

 48                              56                              64
|4A|55|69|42|79|64|57|78|6C|65|67|6F|3D|0A|FF|FF|FF|FF|  |  |  |  |  |  |
  J  U  i  B  y  d  W  x  l  e  g  0  =NL|FF FF FF FF|
```

Offset	Size	Description
0	4	Header signature (ASCII: DFEA – Digital Forensics EDU Archive)
4	1	Lenght of file name (Integer)
5	2	Lenght of data (Little Endian)
7	1	Type of data (Signed Integer) (–1 = ASCII; 0 = base64 encoded)
8	—	Variable file name (ASCII)
9++	—	Data (Binary)
EOF	4	EOF signature (Binary: FF FF FF FF)

2.9 Data, files, context

- Sequence of Bits + Addressing + Interpretation $\rightarrow$ Information
- Where did you find the suspicious data?
  - Binary inside TEMP folder
  - Autorun folder
  - Registry
  - Browser history
  - Command line history

  $\rightarrow$ Data $\rightarrow$ Information $\rightarrow$ Knowledge

- Information $\rightarrow$ Stored in files
- Files $\rightarrow$ Contains data
- Files $\rightarrow$ Data organized in data structures
- Files $\rightarrow$ Meta data describe files
- Files $\rightarrow$ File systems organize files and meta data