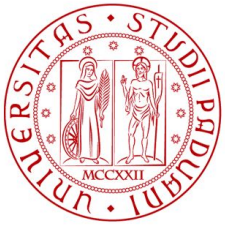
# Ethical Hacking

## Lesson 2: Encoding

Prof. Luca Pajola




UNIVERSITÀ
DEGLI STUDI
DI PADOVA

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

DIPARTIMENTO
MATEMATICA

# Motivations



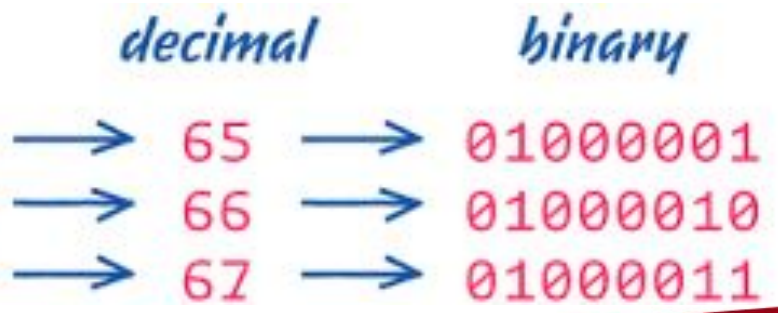
- The world is full of different “environment”
  - e.g., nation, music  notation
- Each “environment” follows its own standard
  - e.g., language, rules
- We need interaction strategies

# Encoding



Why Encoding data?

- **Transform** data so that it can be properly (and safely) consumed by a different type of system
- The goal is not to keep information secret
  - e.g., **keys** are not required
- Encoding can be easily be reversed
  - easy to recognize an encoding strategy
- Decoding is the inverse operation



# Encoding



Some examples of encoding:

- base64
  - web communication
- Hexadecimal
  - simplified version of binary in CS
- Uuencoding
  - binary to text encoding for UNIX

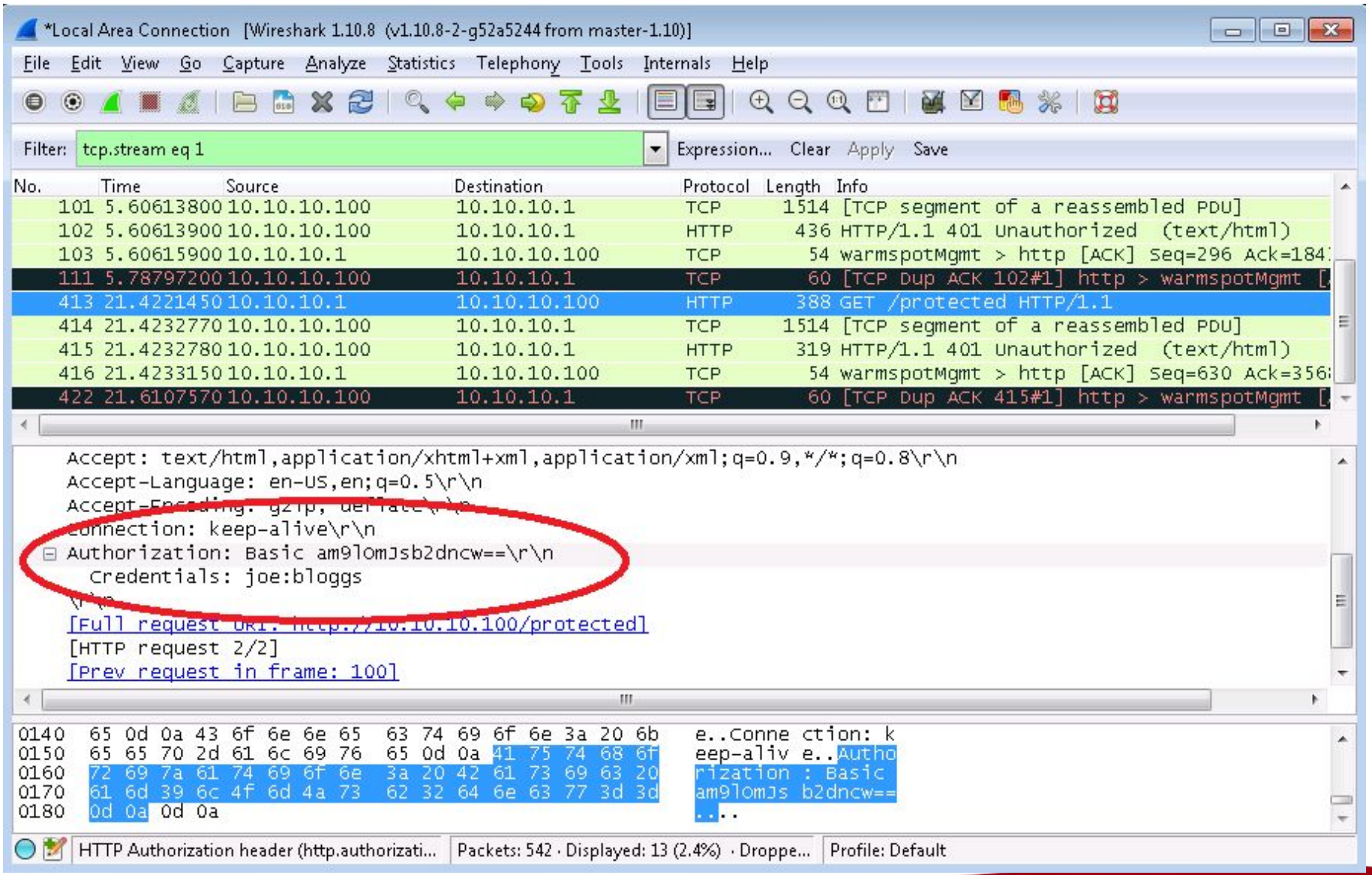
# Base64



- Very common in web communications
- Lengths of final messages **always** a multiple of 4
- Unique alphabet
  - [A-Z, a-z, 0-9, +, /, =]
  - 0 = A, 1 = B, …, 26 = a, 27 = b, …
  - 52 = 0, 53 = 1, …, 62 = +, 63 = /
- Padding
  - Might end with "==" or "="

# Base64




*Local Area Connection [Wireshark 1.10.8 (v1.10.8-2-g52a5244 from master-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: tcp.stream eq 1
Expression... Clear Apply Save
No. Time Source Destination Protocol Length Info
101 5.60613800 10.10.10.100 10.10.10.1 TCP 1514 [TCP segment of a reassembled PDU]
102 5.60613900 10.10.10.100 10.10.10.1 HTTP 436 HTTP/1.1 401 Unauthorized (text/html)
103 5.60615900 10.10.10.1 10.10.10.100 TCP 54 warmspotMgmt > http [ACK] Seq=296 Ack=184
111 5.78797200 10.10.10.100 10.10.10.1 TCP 60 [TCP Dup ACK 102#1] http > warmspotMgmt [
413 21.4221450 10.10.10.1 10.10.10.100 HTTP 388 GET /protected HTTP/1.1
414 21.4232770 10.10.10.100 10.10.10.1 TCP 1514 [TCP segment of a reassembled PDU]
415 21.4232780 10.10.10.100 10.10.10.1 HTTP 319 HTTP/1.1 401 Unauthorized (text/html)
416 21.4233150 10.10.10.1 10.10.10.100 TCP 54 warmspotMgmt > http [ACK] Seq=630 Ack=356
422 21.6107570 10.10.10.100 10.10.10.1 TCP 60 [TCP Dup ACK 415#1] http > warmspotMgmt [
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
Accept-Language: en-US,en;q=0.5\r\n
Connection: keep-alive\r\n
Authorization: Basic am9lomJsb2dncw==\r\n
Credentials: joe:bloggs
[Full request URI: http://10.10.10.100/protected]
[HTTP request 2/2]
[Prev request in frame: 100]
0140 65 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20 6b e..Conne ction: k
0150 65 65 70 2d 61 6c 69 76 65 0d 0a 41 75 74 68 6f eep-aliv e..Autho
0160 72 69 7a 61 74 69 6f 6e 3a 20 42 61 73 69 63 20 rization : Basic
0170 61 6d 39 6c 4f 6d 4a 73 62 32 64 6e 63 77 3d 3d am9lomJs b2dncw==
0180 0d 0a 0d 0a ..
HTTP Authorization header (http.authorizati...
Packets: 542 · Displayed: 13 (2.4%) · Droppe...
Profile: Default

# Base64

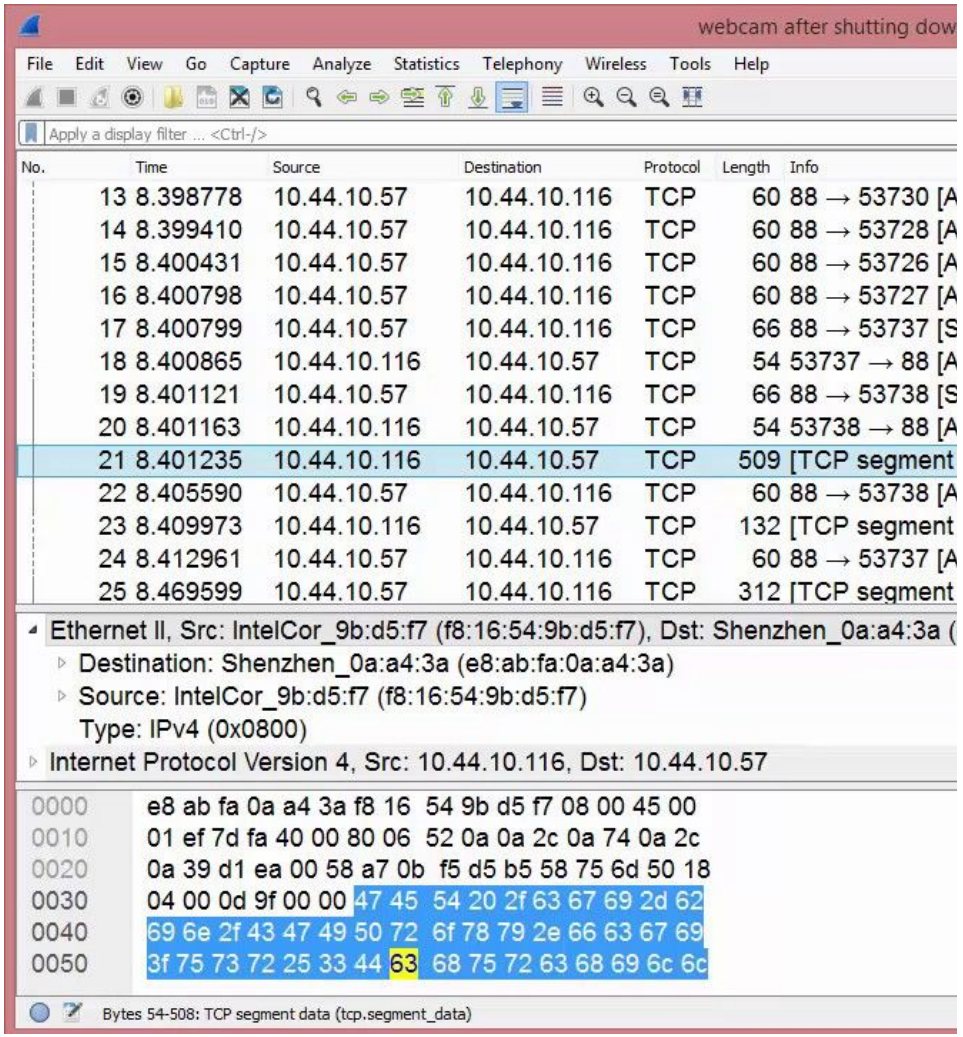


Some examples are:

- pleasure -> cGxlYXN1cmU=
- leasure -> bGVhc3VyZQ==
- easure -> ZWFzdXJl
- asure -> YXN1cmU=
- sure -> c3VyZQ==

# Hexadecimal

- Similar to base64
- The alphabet is [A-F, 0-9]
- Widely used to represent
  - MAC Address
  - Memory dumps



# UUEncoding



- **Always start** with *begin* followed by the *mode* and the *file name*
- **Always end** with both
  - “ ‘ ”
  - *end*

```
begin 600 test.txt
M5&AI<R!I<R!A('1E<W0@9FEL92!F;W(@:6QL=7-T<F%T:6YG('1H92!V87)I
M;W5S"F5N8V]D:6YG(&UE=&AO9',N($QE"=S(&UA:V4@=&AI<R!T97AT(&QO
M;F=E<B!T:&%N"C4W(&)Y=&5S('1O('=R87‘@;&EN97,@=VET:"!"87-E-C0@
E9&%T82P@=&]O+@I'<F5E=&EN9W,L($9R86YK(%!I;&AO9F5R"@‘‘
‘
end
```

# Encoding in Practice



- When we analyze data, it might be represented in an unknown encoding
- How to identify the proper encoding?
  - experience
  - alphabet
  - patterns
  - origin of data

# Questions? Feedback? Suggestions?