securing

Jakub Kałużny

Proprietary network protocols – risky business on the wire

BSides London, 03.06.2015

Who are we

Jakub Kałużny

Sławomir Jasek

Pentesters @ SecuRing

Security assessments of applications, networks, systems...

Agenda

Case studies – proprietary protocols

- Home automation
- Pull printing #1
- Remote desktop
- Pull printing #2
- Trading

Cheatsheet for architects & developers

How to hack it

Proprietary network protocols

- A pentester will encounter one
- Don’t have the protocol specs nor tools to attack it
- How to hack it?
  - decompile the client?
  - search for some tools?
  - watch the raw packets?
- Let’s try!

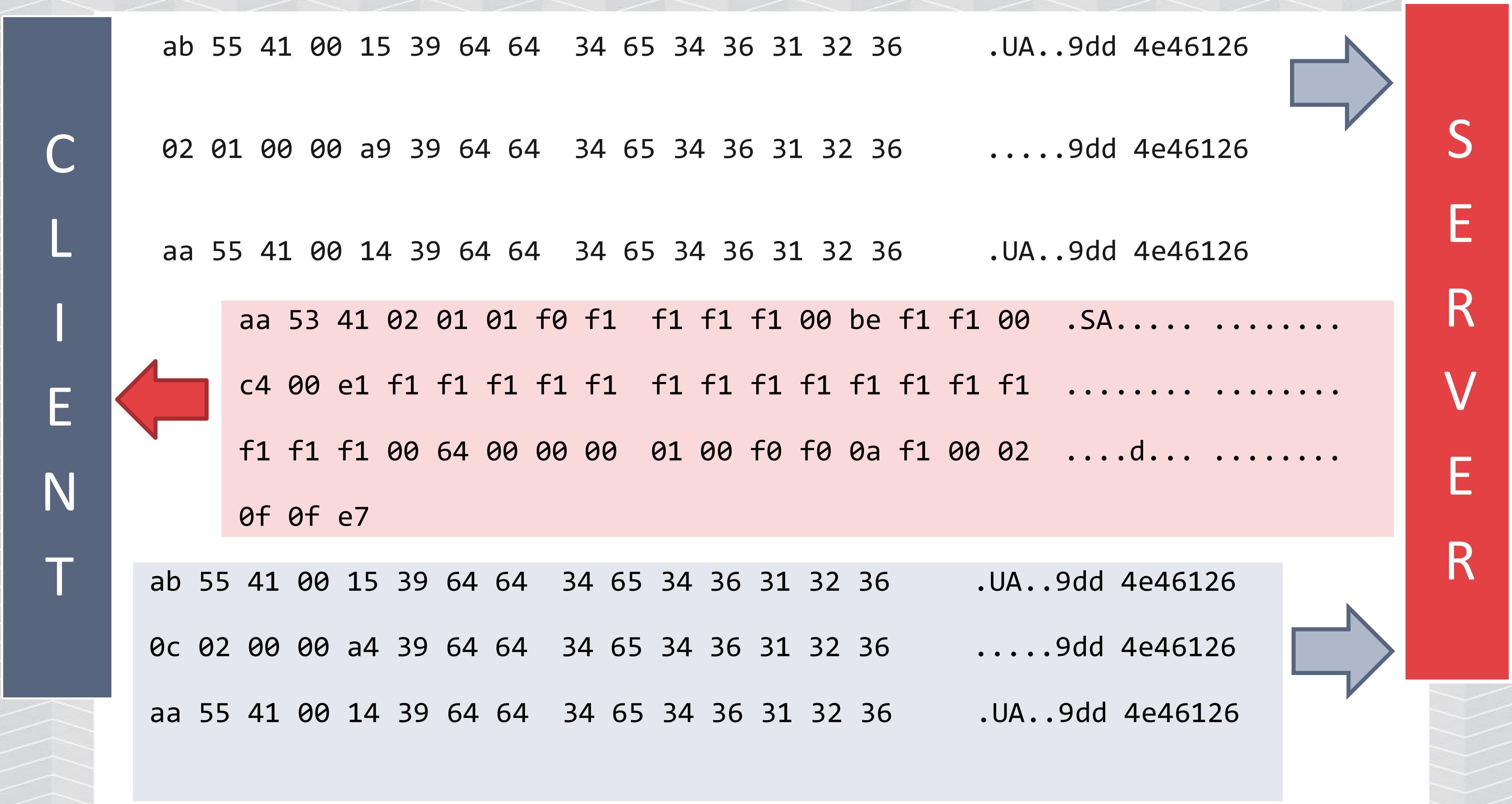
https://www.flickr.com/photos/canonsnapper/2566562866

Home automation remote control

- *„Plug the device, configure your router for port forwarding (and dynamic dns if necessary), set password.”*
- Proprietary TCP protocol, direct connection from Internet to device, password protected access

http://www.flickr.com/photos/99832244@N07/9436065073/

Protocol – a few packets



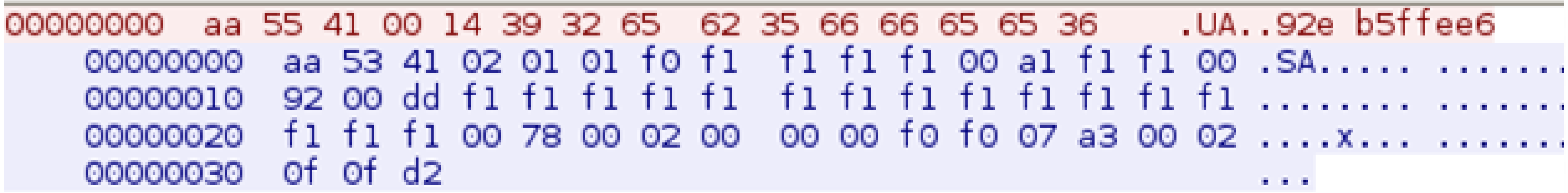
```
ab 55 41 00 15 39 64 64  34 65 34 36 31 32 36     .UA..9dd 4e46126

02 01 00 00 a9 39 64 64  34 65 34 36 31 32 36     .....9dd 4e46126

aa 55 41 00 14 39 64 64  34 65 34 36 31 32 36     .UA..9dd 4e46126
```

```
aa 53 41 02 01 01 f0 f1  f1 f1 f1 00 be f1 f1 00  .SA..... ........
c4 00 e1 f1 f1 f1 f1 f1  f1 f1 f1 f1 f1 f1 f1 f1  ........ ........
f1 f1 f1 00 64 00 00 00  01 00 f0 f0 0a f1 00 02  ....d... ........
0f 0f e7
```

```
ab 55 41 00 15 39 64 64  34 65 34 36 31 32 36     .UA..9dd 4e46126
0c 02 00 00 a4 39 64 64  34 65 34 36 31 32 36     .....9dd 4e46126
aa 55 41 00 14 39 64 64  34 65 34 36 31 32 36     .UA..9dd 4e46126
```

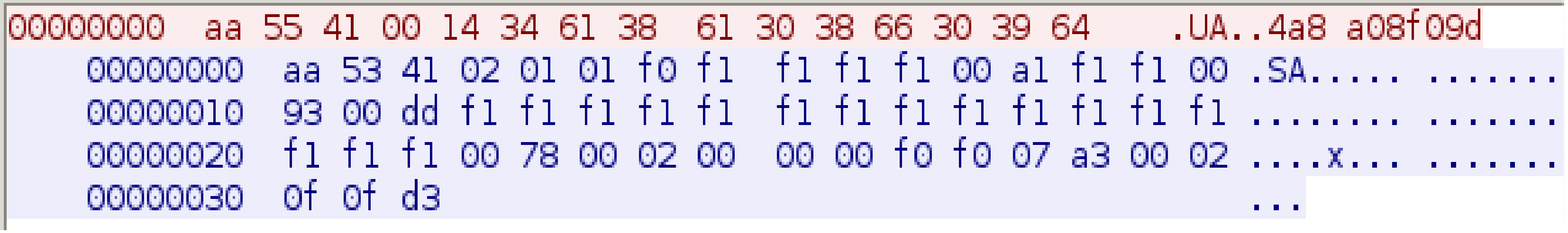
And what if we change the password?

Password 1:



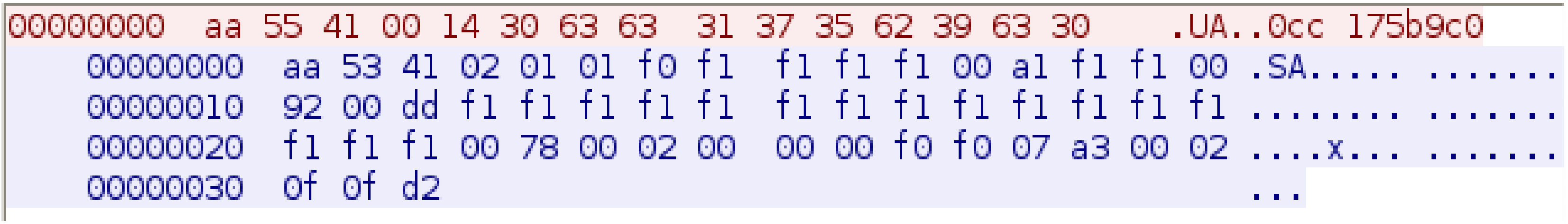
```
00000000  aa 55 41 00 14 39 32 65  62 35 66 66 65 65 36    .UA..92e b5ffee6
    00000000  aa 53 41 02 01 01 f0 f1  f1 f1 f1 00 a1 f1 f1 00 .SA..... .......
    00000010  92 00 dd f1 f1 f1 f1 f1  f1 f1 f1 f1 f1 f1 f1 f1 ........ .......
    00000020  f1 f1 f1 00 78 00 02 00  00 00 f0 f0 07 a3 00 02 ....x... .......
    00000030  0f 0f d2                                         ...
```

Password 2:



```
00000000  aa 55 41 00 14 34 61 38  61 30 38 66 30 39 64    .UA..4a8 a08f09d
    00000000  aa 53 41 02 01 01 f0 f1  f1 f1 f1 00 a1 f1 f1 00 .SA..... .......
    00000010  93 00 dd f1 f1 f1 f1 f1  f1 f1 f1 f1 f1 f1 f1 f1 ........ .......
    00000020  f1 f1 f1 00 78 00 02 00  00 00 f0 f0 07 a3 00 02 ....x... .......
    00000030  0f 0f d3                                         ...
```

Password 3:

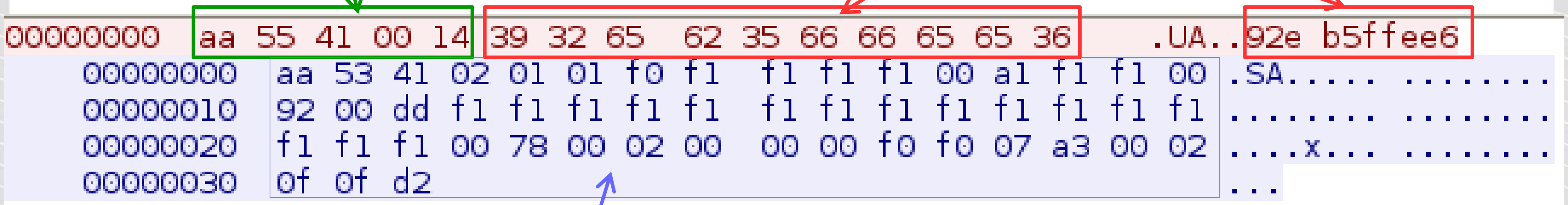


```
00000000  aa 55 41 00 14 30 63 63  31 37 35 62 39 63 30    .UA..0cc 175b9c0
    00000000  aa 53 41 02 01 01 f0 f1  f1 f1 f1 00 a1 f1 f1 00 .SA..... .......
    00000010  92 00 dd f1 f1 f1 f1 f1  f1 f1 f1 f1 f1 f1 f1 f1 ........ .......
    00000020  f1 f1 f1 00 78 00 02 00  00 00 f0 f0 07 a3 00 02 ....x... .......
    00000030  0f 0f d2                                         ...
```

Home automation protocol

internal command (5 bytes)

MD5(password) – first 10 bytes



```
00000000  aa 55 41 00 14 39 32 65  62 35 66 66 65 65 36     .UA..92e b5ffee6
    00000000  aa 53 41 02 01 01 f0 f1  f1 f1 f1 00 a1 f1 f1 00 .SA..... ........
    00000010  92 00 dd f1 f1 f1 f1 f1  f1 f1 f1 f1 f1 f1 f1 f1 ........ ........
    00000020  f1 f1 f1 00 78 00 02 00  00 00 f0 f0 07 a3 00 02 ....x... ........
    00000030  0f 0f d2                                         ...
```

status returned by the appliance (sensors, settings, etc)

Home automation - failures

- Sniffing
- MITM
- Connect directly to the appliance - sniffed hash is enough

- Recommendation: SSL!

Home automation - SSL

Vendor: OK, we have added SSL support!

```
sslcontext=SSLContext.getInstance("TLS");
trustmgr=new TrustManager[1];
trustmgr[0]=new EasyX509TrustManager(null);
sslcontext.init(null, trustmgr, null);
```

- Empty TrustManager – accepts all certificates

Side effect

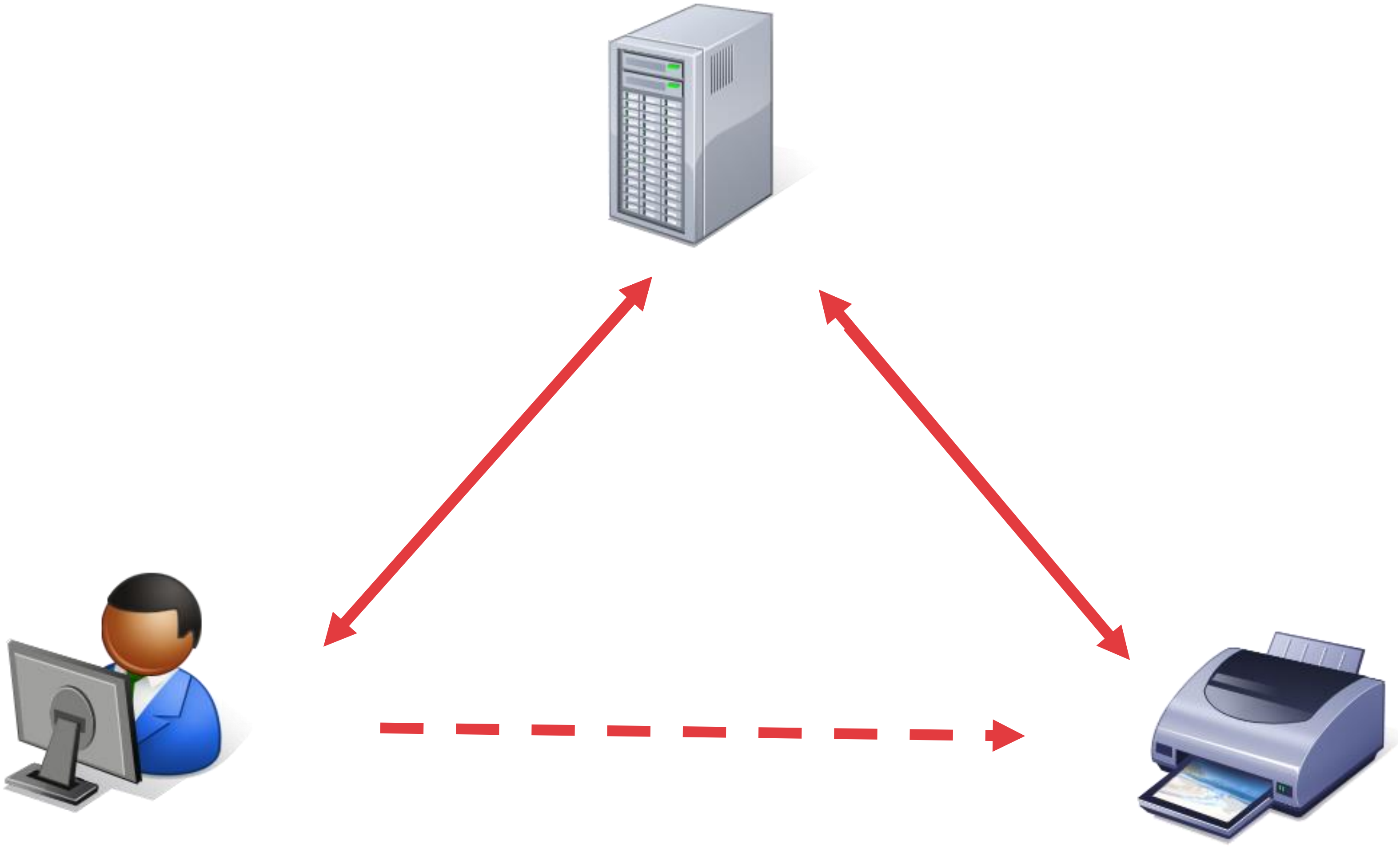
How to build your own appliance:

```
socat tcp4-listen:1234,fork,readbytes=5
/dev/ttyUSB0,min=51
```

And for the new version with SSL support:

```
socat openssl-listen:1234,key=s.key,
cert=s.crt,verify=0,fork,readbytes=5
/dev/ttyUSB0,min=51
```

Pull Printing Solutions



Why hack pull printing?

- Widely used
- Confidential data
- Getting popular

Threat modelling – key risks

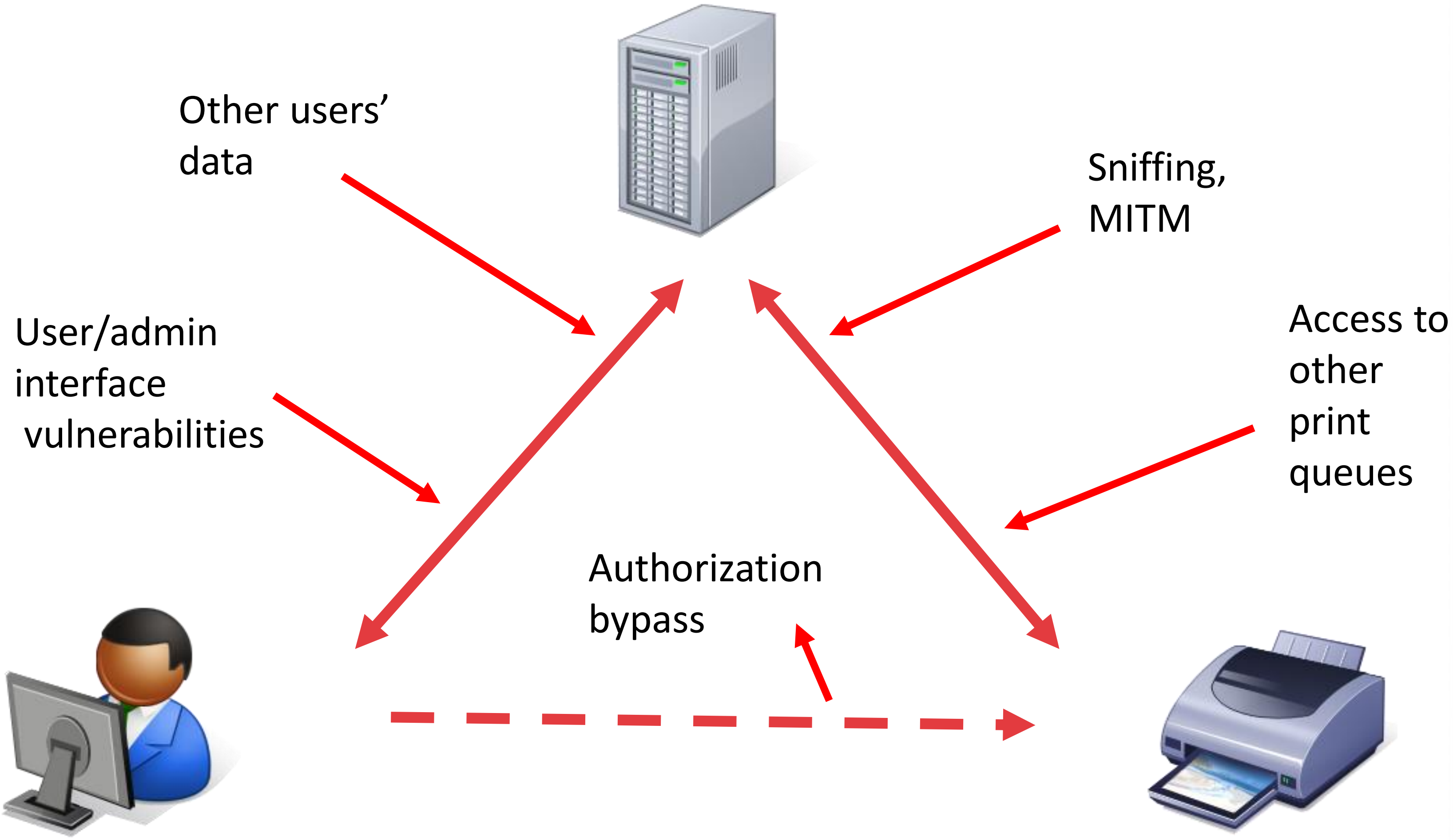
sniffing

print queues

accountability

users’ data

Attack vectors



Other users' data

Sniffing, MITM

User/admin interface vulnerabilities

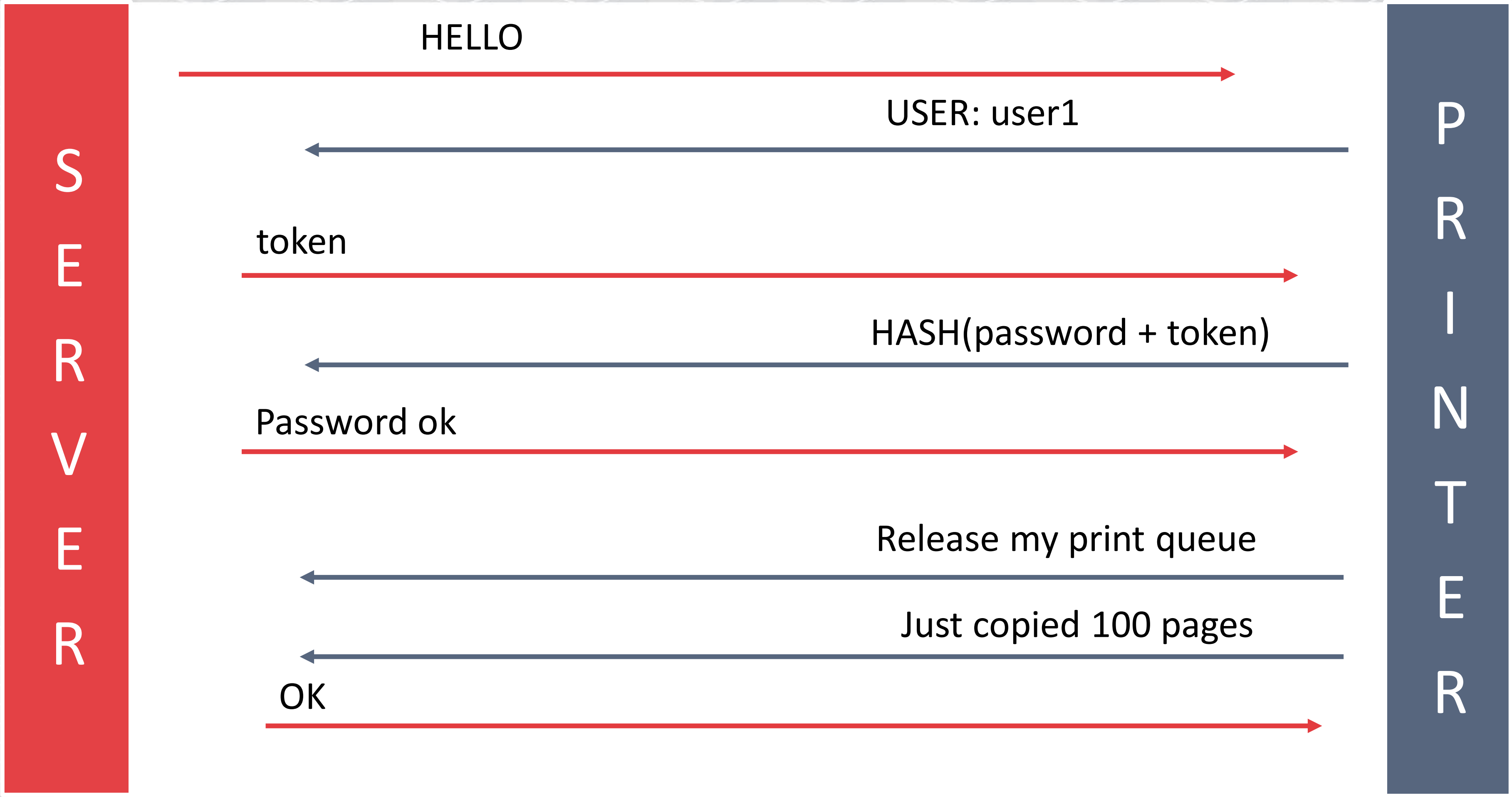
Access to other print queues

Authorization bypass

Pull Printing #1 – access control

“**Secure print release** (…) can integrate card-swipe user authentication at devices (…) ensuring jobs are **only** printed when the collecting user is present.”

Pull Printing #1 – binary protocol



SERVER

PRINTER

HELLO

USER: user1

token

HASH(password + token)

Password ok

Release my print queue

Just copied 100 pages

OK

Pull Printing #1 – closer look

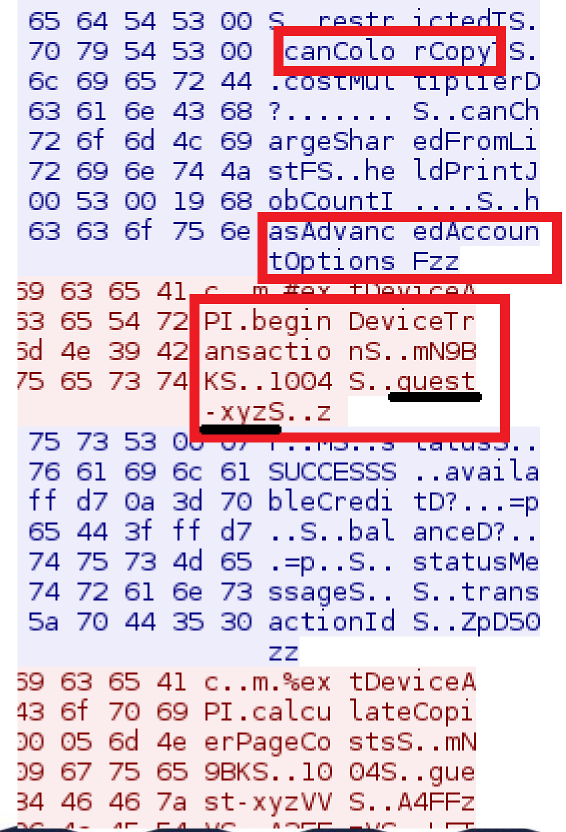
S
E
R
V
E
R

Release print queue for user “guest-xyz”

Release my print queue

Charge user “guest-xyz” for copying 100 pages

Just copied 100 pages



```
65 64 54 53 00 S..restr ictedTS.
70 79 54 53 00 canColo rCopy S.
6c 69 65 72 44 .costMul tiplierD
63 61 6e 43 68 ?....... S..canCh
72 6f 6d 4c 69 argeShar edFromLi
72 69 6e 74 4a stFS..he ldPrintJ
00 53 00 19 68 obCountI ....S..h
63 63 6f 75 6e asAdvanc edAccoun
               tOptions Fzz
59 63 65 41 c..m.#ex tDeviceA
53 65 54 72 PI.begin DeviceTr
5d 4e 39 42 ansactio nS..mN9B
75 65 73 74 KS..1004 S..guest
            -xyzS..z
75 73 53 00 07 ...MS..S tatusS..
76 61 69 6c 61 SUCCESSS ..availa
ff d7 0a 3d 70 bleCredi tD?...=p
65 44 3f ff d7 ..S..bal anceD?..
74 75 73 4d 65 .=p..S.. statusMe
74 72 61 6e 73 ssageS.. S..trans
5a 70 44 35 30 actionId S..ZpD50
               zz
59 63 65 41 c..m.%ex tDeviceA
43 6f 70 69 PI.calcu lateCopi
00 05 6d 4e erPageCo stsS..mN
09 67 75 65 9BKS..10 04S..gue
34 46 46 7a st-xyzVV S..A4FFz
```

User permissions

beginDeviceTransaction
(…) guest-xyz

P
R
I
N
T
E
R

Pull printing #1 - consequences

sniffing

print queues

accountability

users’ data

Pull printing #1 - vendor gets notified

- Gave access to KB and support service
- And all versions of software
- Responded in few hours and patched in few days
- Was happy to be pentested

Remote desktop protocol

X-win „on steroids” (encryption, compression, access control...)

Mainframe access for critical business operations

„More than 100,000 users around the world”

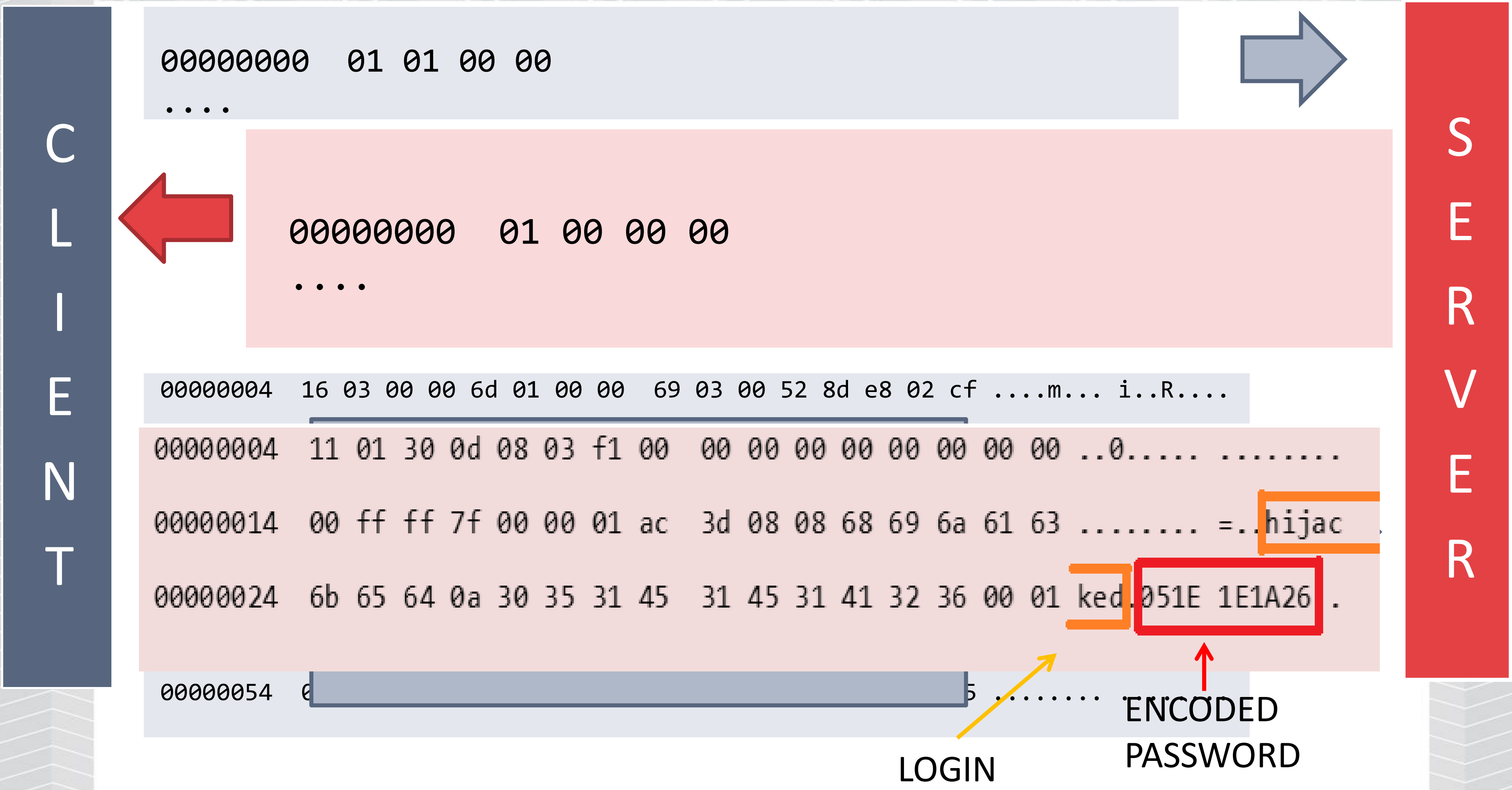
„Prevents unauthorized eavesdropping

FIPS 140-2 Validated

End-to-end data encryption”

http://en.wikipedia.org/wiki/X_Window_System_protocols_and_architecture

Remote desktop protocol



```
00000000  01 01 00 00
....
```

```
00000000  01 00 00 00
....
```

```
00000004  16 03 00 00 6d 01 00 00  69 03 00 52 8d e8 02 cf ....m... i..R....
00000004  11 01 30 0d 08 03 f1 00  00 00 00 00 00 00 00 00 ..0..... ........
00000014  00 ff ff 7f 00 00 01 ac  3d 08 08 68 69 6a 61 63 ........ =..hijac
00000024  6b 65 64 0a 30 35 31 45  31 45 31 41 32 36 00 01 ked.051E 1E1A26 .
00000054  0[redacted]5 ........
```

Password

TestingPassword1234TestingPassword

54657374696e6750617373776f726431323334546573696e6750617373776f7264

XOR

1c101e1900000032080117572c1d095c475d5d3704071d060014702d1a1e1e1b1700

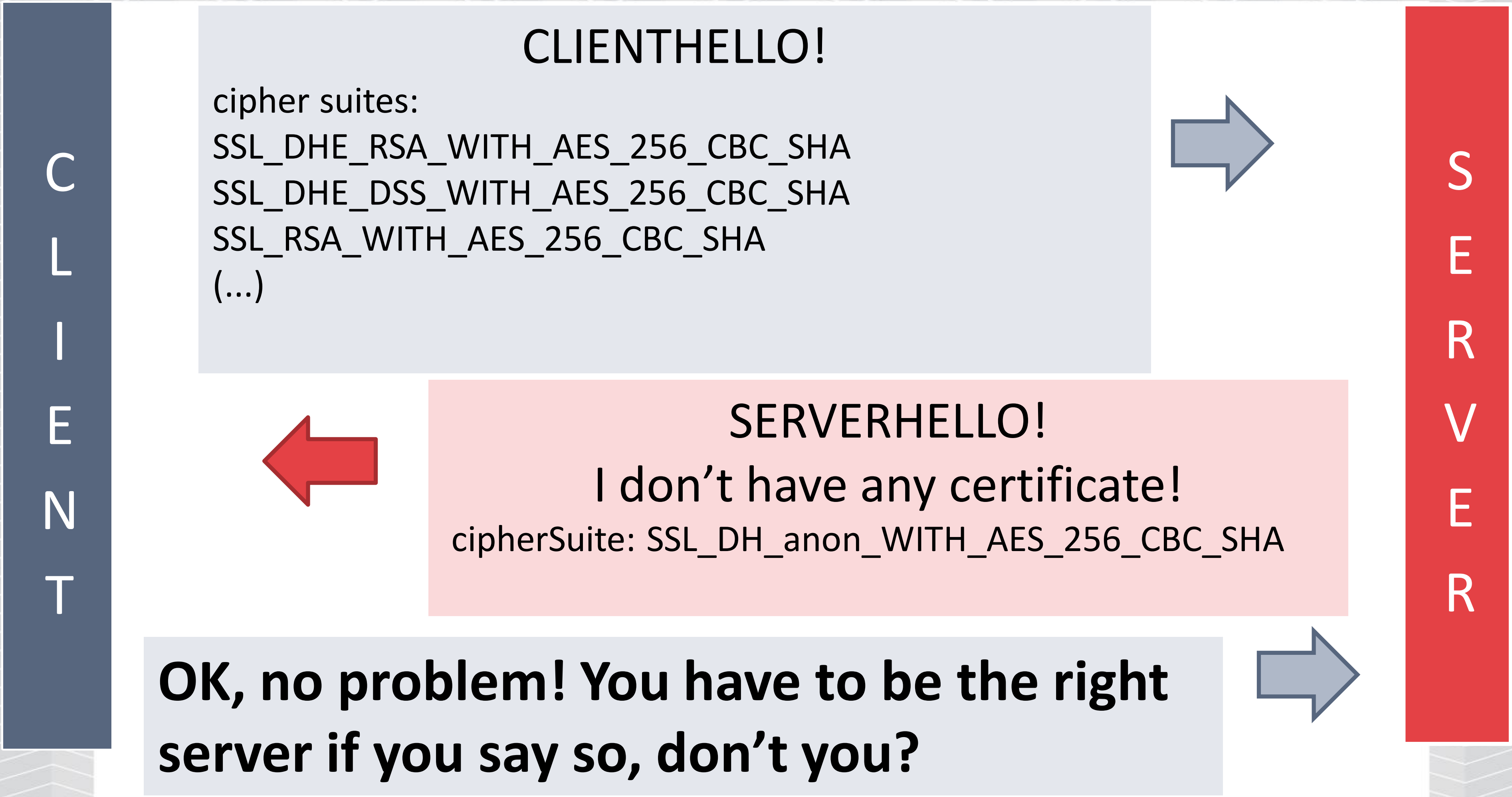
=

48756d6d696e67626972642043 6f6d6d756e69636174696f6e73204c696d69746564

[redacted] Communications Limited

Remote Desktop - SSL

default configuration



CLIENT

CLIENTHELLO!

cipher suites:
SSL_DHE_RSA_WITH_AES_256_CBC_SHA
SSL_DHE_DSS_WITH_AES_256_CBC_SHA
SSL_RSA_WITH_AES_256_CBC_SHA
(...)

SERVERHELLO!

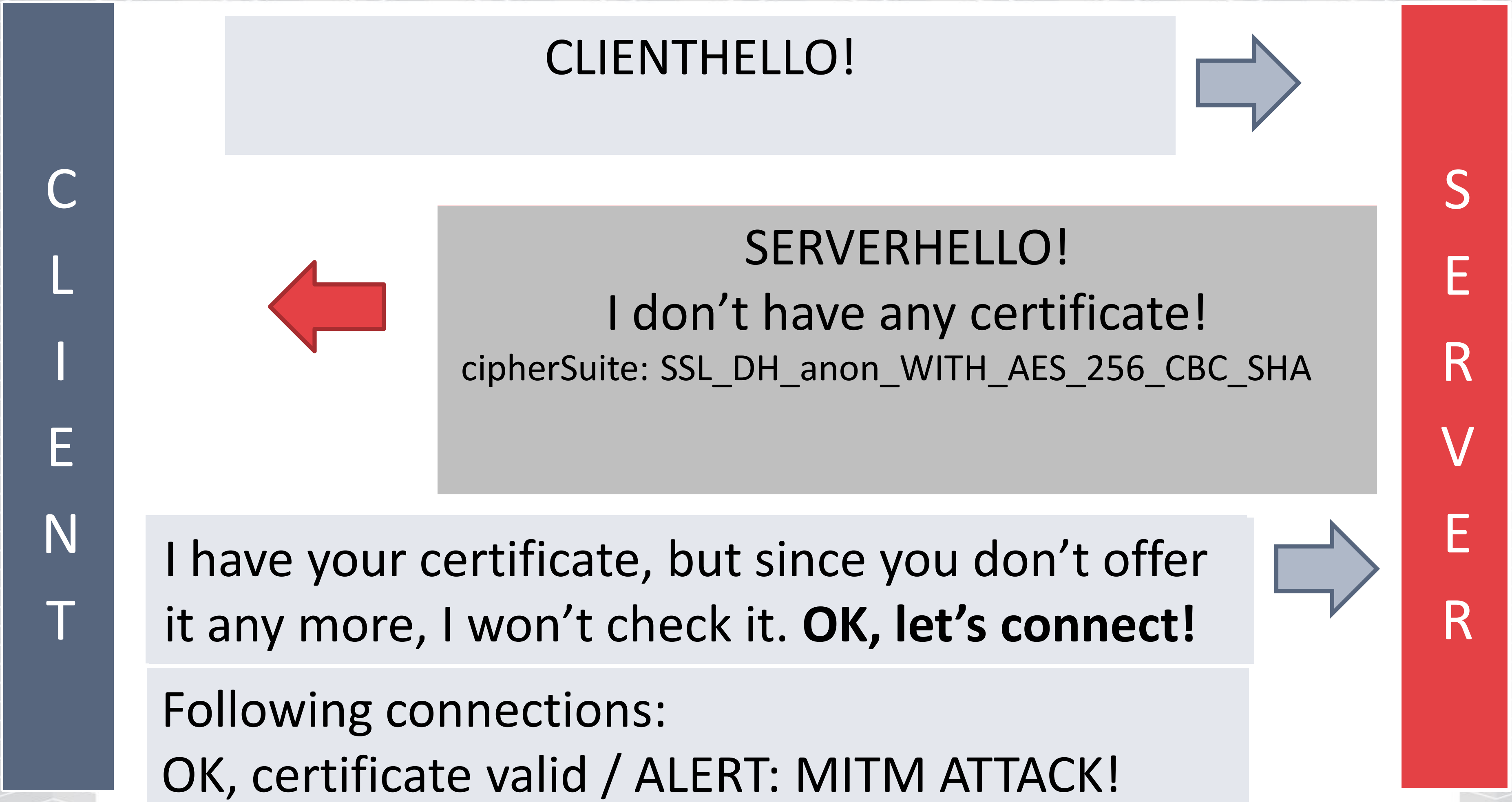
I don’t have any certificate!

cipherSuite: SSL_DH_anon_WITH_AES_256_CBC_SHA

OK, no problem! You have to be the right server if you say so, don’t you?

SERVER

Remote Desktop - SSL certificates configured



CLIENT

SERVER

CLIENTHELLO!

SERVERHELLO!
I don’t have any certificate!
cipherSuite: SSL_DH_anon_WITH_AES_256_CBC_SHA

I have your certificate, but since you don’t offer it any more, I won’t check it. **OK, let’s connect!**

Following connections:
OK, certificate valid / ALERT: MITM ATTACK!

Remote desktop protocol - vendor

„We don’t know PGP, use zip with our CEO’s name as password”

Do not plan to solve the issues (?)

```
> /dev/null 2>&1
```

Full disclosure!

... and a few weeks later the mysterious shut down of our beloved ;)

Pull Printing #2 - encryption

“is a modern printing solution that **safeguards document confidentiality** and unauthorized access to print, scan, copy and e-mail functions. Its user-authentication **provides air-tight security** on your shared MFPs that function as personal printers.”

Vendor ensures

„Documents are delivered **only** into the right hands”

„Information is kept **confidential**. **No risk** of being left unattended at the printer”

„Document collection is **safe anytime and anywhere** — no “print and sprint”.”

„Integration with other enterprise applications and workflows **is kept secure** through single sign-on”

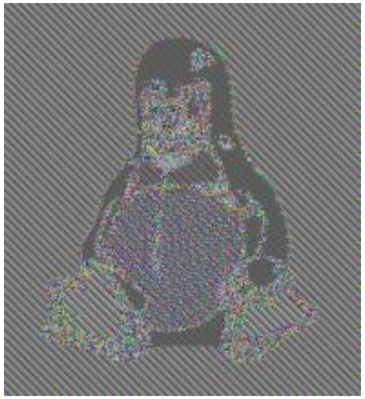
Pull Printing #2 – binary protocol

First look on communication:

- TCP, 2 ports
- No cleartext, no SSL
- Seems to follow some scheme…

Ex1: Deeper sight on traffic

```
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6..............:.c..<..~7%../}W...}....A1.......dh{hR9....
.d.._.]..y.{@...l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........a.`OVT...9/.;.s.
....q...~+..@.9.'.o.]...b..o..
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6...........l
..6............f...b...>..}U..4-{.K./o%#..;..l
..6...........l
..6...........l
..6............l)........y...^Ql
..6..........r\...ix......0.n...;4.o.....p\..gSd............qg........
+).q8...z..F..b1.Fr...l
..6...........l
..6...........l
..6...........l
..6...........l
```



https://en.wikipedia.org/wiki/ECB_mode

Pull Printing #2 - Reverse-engineered

- Hardcoded RSA certificate in printer embedded software
- No trust store
- AES-128 ECB used for traffic encryption
- Same protocol in admin interface

Pull Printing #2 - Consequences

sniffing

print queues

accountability

users’ data

Pull Printing #1 - vendor gets notified

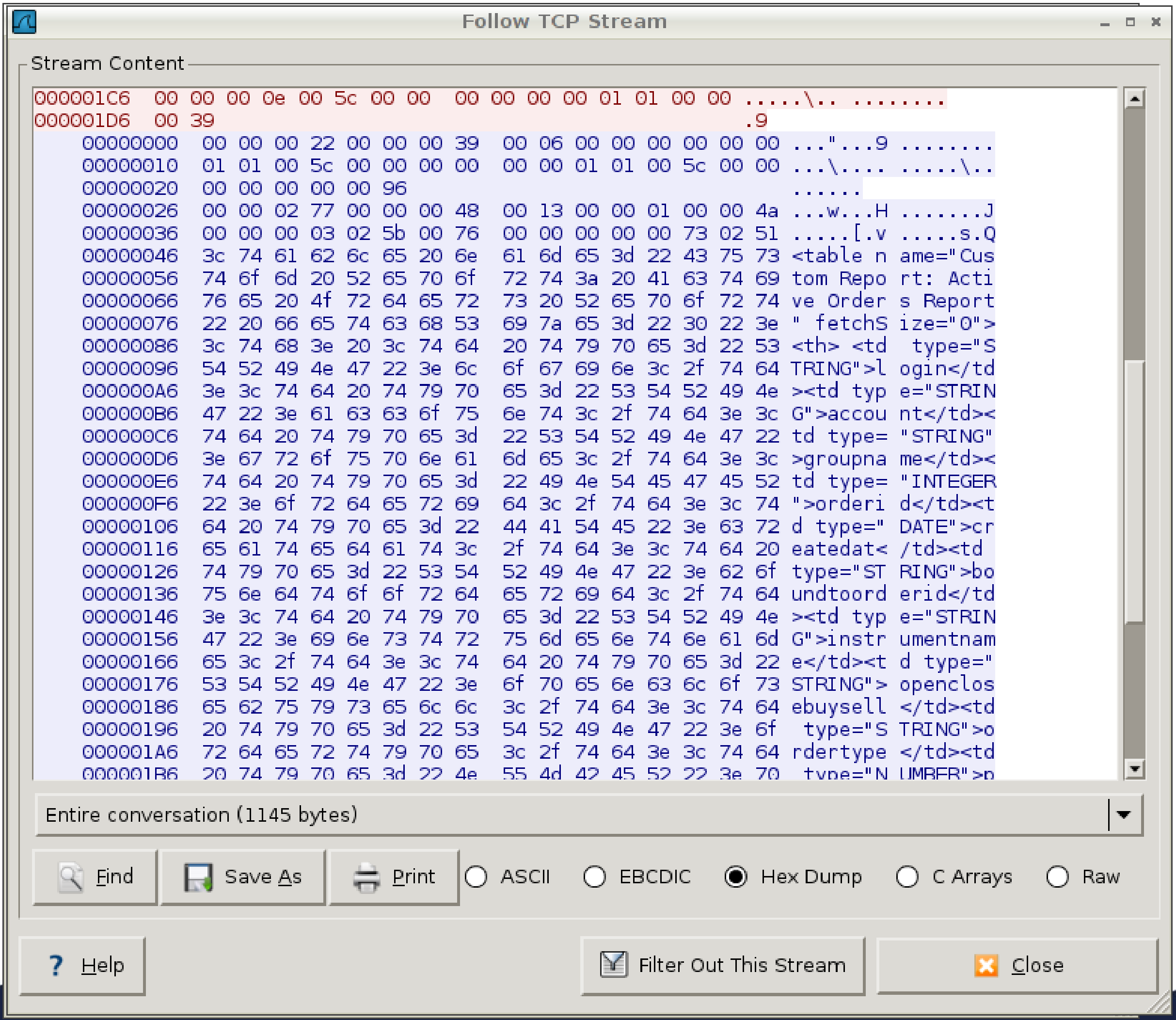
"(…) system has been deployed at many high security customers and **has passed internal audits**."

Trading protocol

- An online application for instant financial operations
- A proprietary, binary protocol, designed in order to minimise delays
- TCP in SSL tunnel

https://www.flickr.com/photos/tradingrichmom/5571144428/

Trading protocol

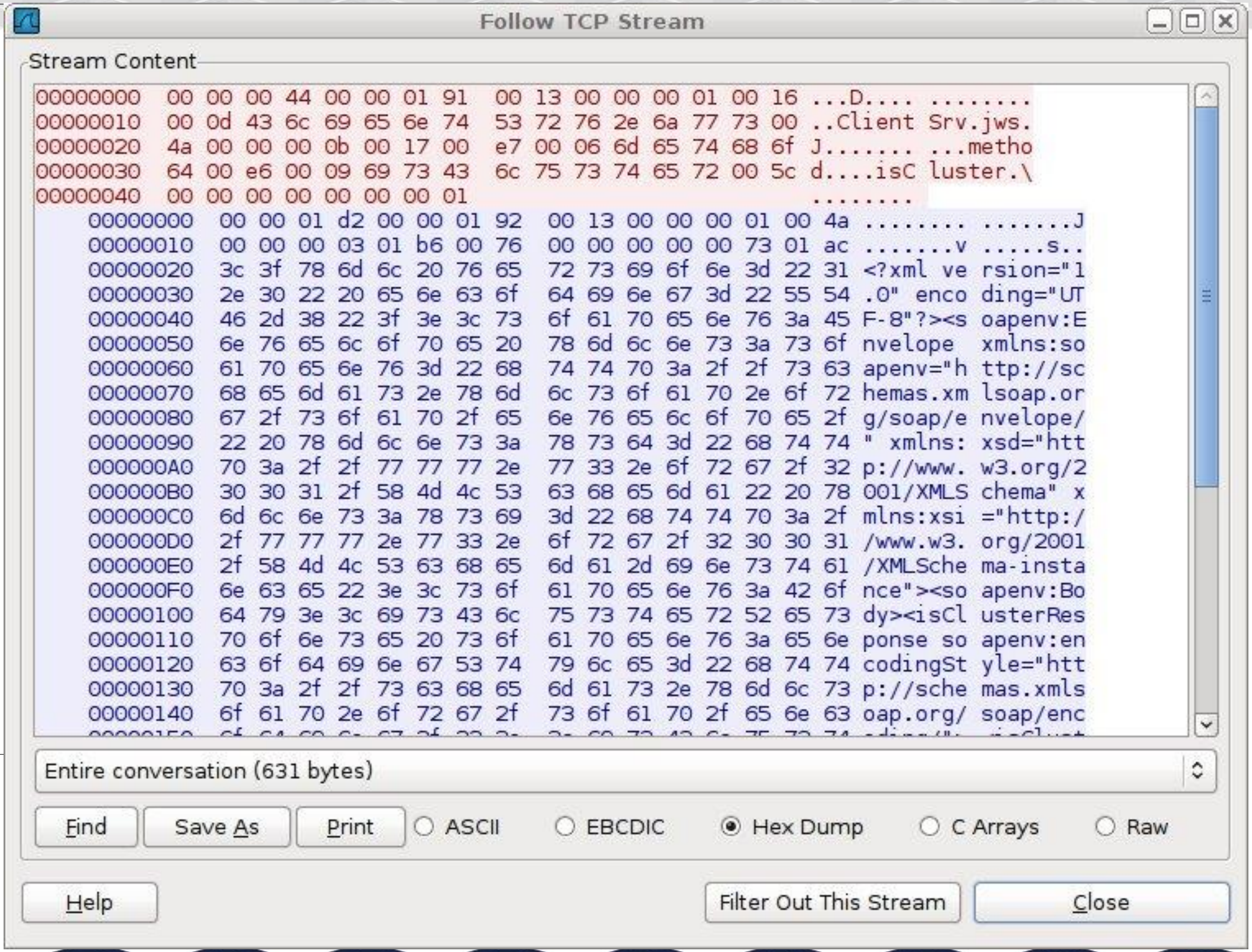


```
000001C6  00 00 00 0e 00 5c 00 00  00 00 00 00 01 01 00 00 .....\.. ........
000001D6  00 39                                            .9
    00000000  00 00 00 22 00 00 00 39  00 06 00 00 00 00 00 00 ..."...9 ........
    00000010  01 01 00 5c 00 00 00 00  00 00 01 01 00 5c 00 00 ...\.... .....\..
    00000020  00 00 00 00 00 96                                ......
    00000026  00 00 02 77 00 00 00 48  00 13 00 00 01 00 00 4a ...w...H .......J
    00000036  00 00 00 03 02 5b 00 76  00 00 00 00 00 73 02 51 .....[.v .....s.Q
    00000046  3c 74 61 62 6c 65 20 6e  61 6d 65 3d 22 43 75 73 <table n ame="Cus
    00000056  74 6f 6d 20 52 65 70 6f  72 74 3a 20 41 63 74 69 tom Repo rt: Acti
    00000066  76 65 20 4f 72 64 65 72  73 20 52 65 70 6f 72 74 ve Order s Report
    00000076  22 20 66 65 74 63 68 53  69 7a 65 3d 22 30 22 3e " fetchS ize="0">
    00000086  3c 74 68 3e 20 3c 74 64  20 74 79 70 65 3d 22 53 <th> <td  type="S
    00000096  54 52 49 4e 47 22 3e 6c  6f 67 69 6e 3c 2f 74 64 TRING">l ogin</td
    000000A6  3e 3c 74 64 20 74 79 70  65 3d 22 53 54 52 49 4e ><td typ e="STRIN
    000000B6  47 22 3e 61 63 63 6f 75  6e 74 3c 2f 74 64 3e 3c G">accou nt</td><
    000000C6  74 64 20 74 79 70 65 3d  22 53 54 52 49 4e 47 22 td type= "STRING"
    000000D6  3e 67 72 6f 75 70 6e 61  6d 65 3c 2f 74 64 3e 3c >groupna me</td><
    000000E6  74 64 20 74 79 70 65 3d  22 49 4e 54 45 47 45 52 td type= "INTEGER
    000000F6  22 3e 6f 72 64 65 72 69  64 3c 2f 74 64 3e 3c 74 ">orderi d</td><t
    00000106  64 20 74 79 70 65 3d 22  44 41 54 45 22 3e 63 72 d type=" DATE">cr
    00000116  65 61 74 65 64 61 74 3c  2f 74 64 3e 3c 74 64 20 eatedat< /td><td
    00000126  74 79 70 65 3d 22 53 54  52 49 4e 47 22 3e 62 6f type="ST RING">bo
    00000136  75 6e 64 74 6f 6f 72 64  65 72 69 64 3c 2f 74 64 undtoord erid</td
    00000146  3e 3c 74 64 20 74 79 70  65 3d 22 53 54 52 49 4e ><td typ e="STRIN
    00000156  47 22 3e 69 6e 73 74 72  75 6d 65 6e 74 6e 61 6d G">instr umentnam
    00000166  65 3c 2f 74 64 3e 3c 74  64 20 74 79 70 65 3d 22 e</td><t d type="
    00000176  53 54 52 49 4e 47 22 3e  6f 70 65 6e 63 6c 6f 73 STRING"> openclos
    00000186  65 62 75 79 73 65 6c 6c  3c 2f 74 64 3e 3c 74 64 ebuysell </td><td
    00000196  20 74 79 70 65 3d 22 53  54 52 49 4e 47 22 3e 6f  type="S TRING">o
    000001A6  72 64 65 72 74 79 70 65  3c 2f 74 64 3e 3c 74 64 rdertype </td><td
    000001B6  20 74 79 70 65 3d 22 4e  55 4d 42 45 52 22 3e 70  type="N UMBER">p
```

Entire conversation (1145 bytes)

That’s interesting!

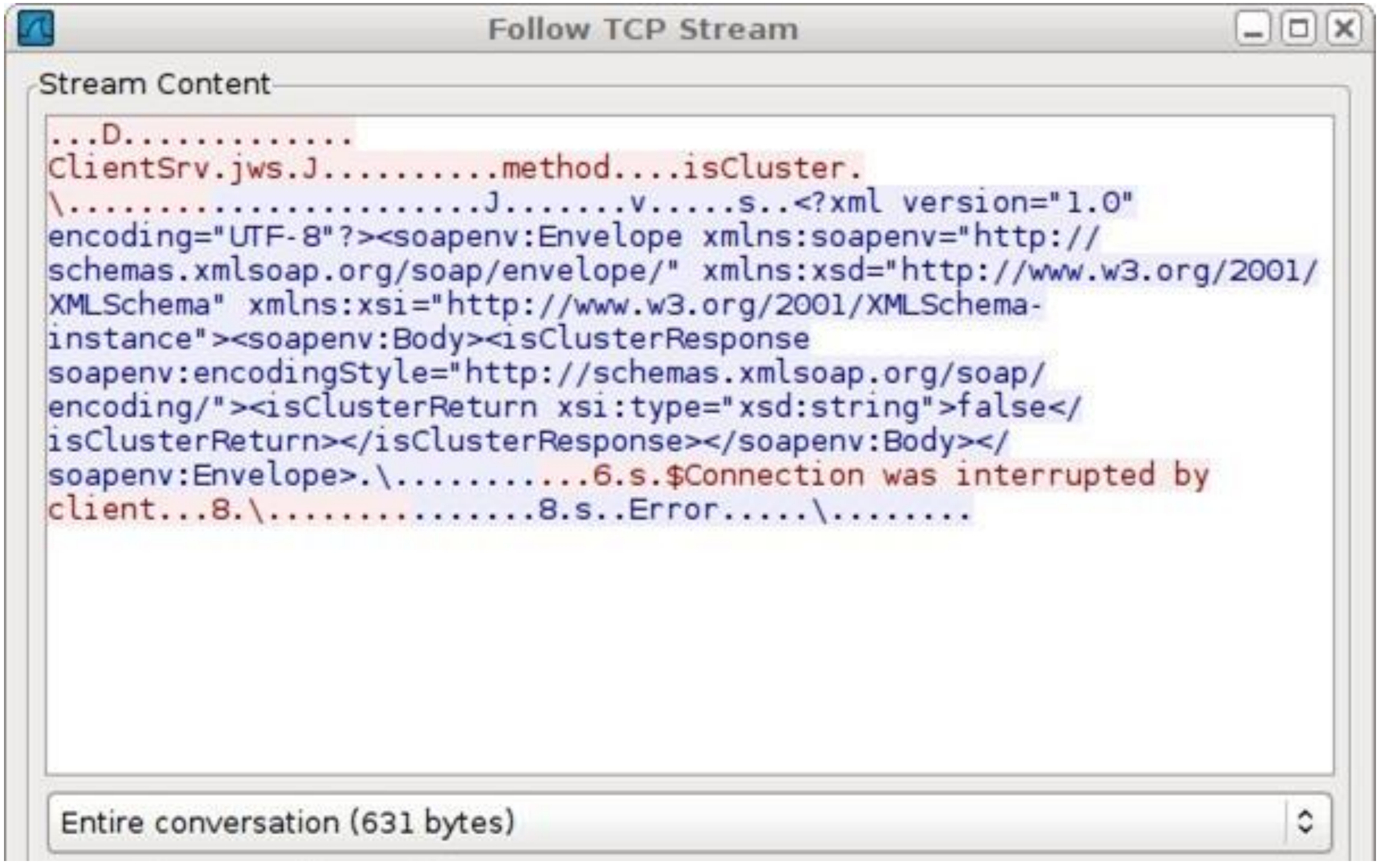
The picture can't be displayed.



```
00000000  00 00 00 44 00 00 01 91  00 13 00 00 00 01 00 16 ...D.... ........
00000010  00 0d 43 6c 69 65 6e 74  53 72 76 2e 6a 77 73 00 ..Client Srv.jws.
00000020  4a 00 00 00 0b 00 17 00  e7 00 06 6d 65 74 68 6f J....... ...metho
00000030  64 00 e6 00 09 69 73 43  6c 75 73 74 65 72 00 5c d....isC luster.\
00000040  00 00 00 00 00 00 00 01                          ........
    00000000  00 00 01 d2 00 00 01 92  00 13 00 00 00 01 00 4a ........ .......J
    00000010  00 00 00 03 01 b6 00 76  00 00 00 00 00 73 01 ac .......v .....s..
    00000020  3c 3f 78 6d 6c 20 76 65  72 73 69 6f 6e 3d 22 31 <?xml ve rsion="1
    00000030  2e 30 22 20 65 6e 63 6f  64 69 6e 67 3d 22 55 54 .0" enco ding="UT
    00000040  46 2d 38 22 3f 3e 3c 73  6f 61 70 65 6e 76 3a 45 F-8"?><s oapenv:E
    00000050  6e 76 65 6c 6f 70 65 20  78 6d 6c 6e 73 3a 73 6f nvelope  xmlns:so
    00000060  61 70 65 6e 76 3d 22 68  74 74 70 3a 2f 2f 73 63 apenv="h ttp://sc
    00000070  68 65 6d 61 73 2e 78 6d  6c 73 6f 61 70 2e 6f 72 hemas.xm lsoap.or
    00000080  67 2f 73 6f 61 70 2f 65  6e 76 65 6c 6f 70 65 2f g/soap/e nvelope/
    00000090  22 20 78 6d 6c 6e 73 3a  78 73 64 3d 22 68 74 74 " xmlns: xsd="htt
    000000A0  70 3a 2f 2f 77 77 77 2e  77 33 2e 6f 72 67 2f 32 p://www. w3.org/2
    000000B0  30 30 31 2f 58 4d 4c 53  63 68 65 6d 61 22 20 78 001/XMLS chema" x
    000000C0  6d 6c 6e 73 3a 78 73 69  3d 22 68 74 74 70 3a 2f mlns:xsi ="http:/
    000000D0  2f 77 77 77 2e 77 33 2e  6f 72 67 2f 32 30 30 31 /www.w3. org/2001
    000000E0  2f 58 4d 4c 53 63 68 65  6d 61 2d 69 6e 73 74 61 /XMLSche ma-insta
    000000F0  6e 63 65 22 3e 3c 73 6f  61 70 65 6e 76 3a 42 6f nce"><so apenv:Bo
    00000100  64 79 3e 3c 69 73 43 6c  75 73 74 65 72 52 65 73 dy><isCl usterRes
    00000110  70 6f 6e 73 65 20 73 6f  61 70 65 6e 76 3a 65 6e ponse so apenv:en
    00000120  63 6f 64 69 6e 67 53 74  79 6c 65 3d 22 68 74 74 codingSt yle="htt
    00000130  70 3a 2f 2f 73 63 68 65  6d 61 73 2e 78 6d 6c 73 p://sche mas.xmls
    00000140  6f 61 70 2e 6f 72 67 2f  73 6f 61 70 2f 65 6e 63 oap.org/ soap/enc
```

Entire conversation (631 bytes)

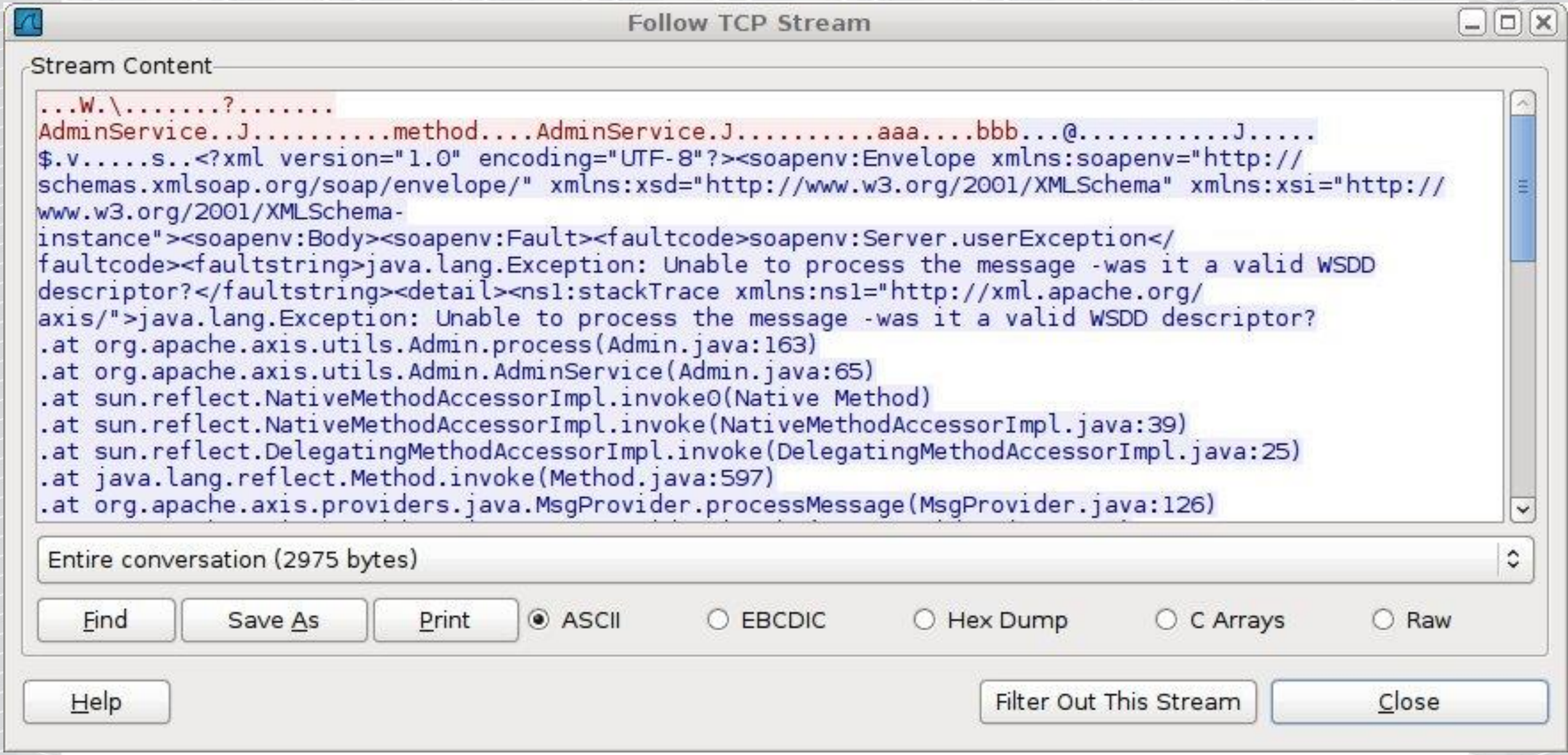
That's interesting!



```
...D.............
ClientSrv.jws.J.........method....isCluster.
\......................J.......v.....s..<?xml version="1.0"
encoding="UTF-8"?><soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/
XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"><soapenv:Body><isClusterResponse
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/"><isClusterReturn xsi:type="xsd:string">false</
isClusterReturn></isClusterResponse></soapenv:Body></
soapenv:Envelope>.\..........6.s.$Connection was interrupted by
client...8.\...............8.s..Error.....\........
```

Entire conversation (631 bytes)

And what if we...

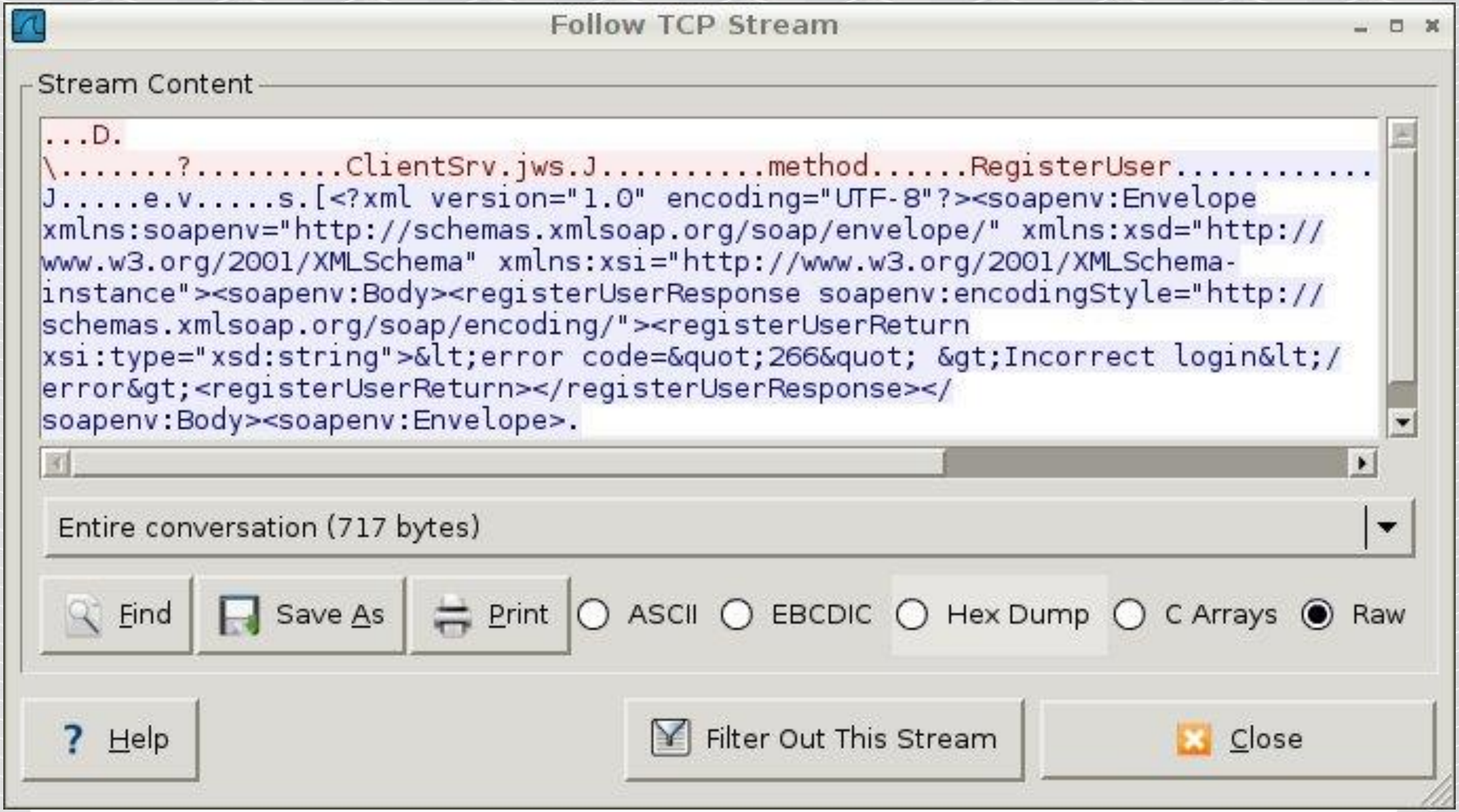


```
...W.\.......?.......
AdminService..J..........method....AdminService.J..........aaa....bbb...@...........J.....
$.v.....s..<?xml version="1.0" encoding="UTF-8"?><soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-
instance"><soapenv:Body><soapenv:Fault><faultcode>soapenv:Server.userException</
faultcode><faultstring>java.lang.Exception: Unable to process the message -was it a valid WSDD
descriptor?</faultstring><detail><ns1:stackTrace xmlns:ns1="http://xml.apache.org/
axis/">java.lang.Exception: Unable to process the message -was it a valid WSDD descriptor?
.at org.apache.axis.utils.Admin.process(Admin.java:163)
.at org.apache.axis.utils.Admin.AdminService(Admin.java:65)
.at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
.at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)
.at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)
.at java.lang.reflect.Method.invoke(Method.java:597)
.at org.apache.axis.providers.java.MsgProvider.processMessage(MsgProvider.java:126)
```

Entire conversation (2975 bytes)

● ASCII ○ EBCDIC ○ Hex Dump ○ C Arrays ○ Raw

And how about...



```
...D.
\.......?.........ClientSrv.jws.J..........method......RegisterUser...........
J.....e.v.....s.[<?xml version="1.0" encoding="UTF-8"?><soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"><soapenv:Body><registerUserResponse soapenv:encodingStyle="http://
schemas.xmlsoap.org/soap/encoding/"><registerUserReturn
xsi:type="xsd:string">&lt;error code=&quot;266&quot; &gt;Incorrect login&lt;/
error&gt;<registerUserReturn></registerUserResponse></
soapenv:Body><soapenv:Envelope>.
```

Entire conversation (717 bytes)

○ ASCII ○ EBCDIC ○ Hex Dump ○ C Arrays ● Raw

RegisterUser

```
<soapenv:Body> <registerUserResponse
   soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encodin
   g/">

<registerUserReturn xsi:type="xsd:string">

    &lt;error code=&quot;266&quot; &gt;Incorrect
       login&lt;/error&gt;

    </registerUserReturn></registerUserResponse></soapenv:Body>
```

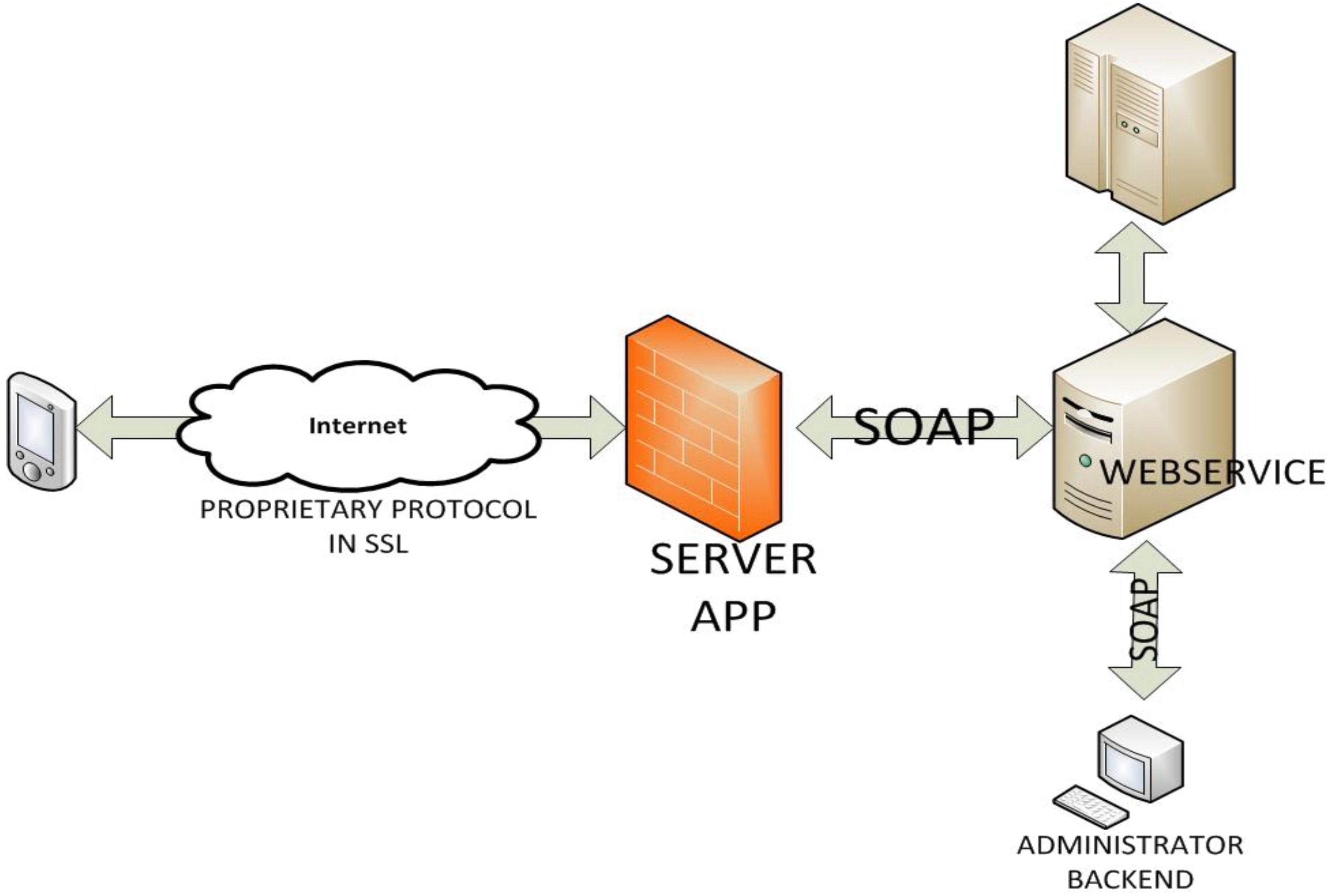
- Incorrect password
- Incorrect first name
- Group with name null doesn't exist
- Group with name admin doesn't exist
- Group with name Administrator doesn't exist
- And how about „root”?

Game Over

```
<soapenv:Body>

 <registerUserResponse
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/">

<registerUserReturn xsi:type="xsd:string">

User was registered sucessfully with id=5392745
```

So now we can manage all the other accounts and spend their money!

Architecture



Cheat sheet – owners

While deploying a proprietary solution:

- Get it pentested
- Verify vendor claims
- Ask the vendor for secure development lifecycle, procedures of addressing vulnerabilities, previous bugs

Cheat sheet - developers

Protocol is NOT secure by its secrecy

Proper encryption. Use known standards, implement them with care.

Input validation, access control, many layers of security, least privilege principle...

Beware backwards compatibility

How to hack protocols?

Decompile client?

Inject code?

Search for the specs?

Use some tools?

Watch the packets?

https://www.flickr.com/photos/nkphillips/2865781749

Look for the fine manual

- There may be unofficial client, or e.g. wireshark plugin
- Ask for the docs ☺
- Search for them
- Yes, we have found internal protocol specification by google hacking!

Decompile client

Sometimes easy – e.g. not obfuscated Android application:

```
byte abyte3[] = pass.getBytes();
byte abyte4[] =
MessageDigest.getInstance("MD5").digest
(abyte3);
```

Sometimes really hard & time consuming.

May be fun, but often leads astray

Watch the packets

Various tools to analyze proprietary protocols

time consuming, usually do not work

Raw, just try to spot some scheme

of course with a little help of your friends: wireshark, tcpdump, ssldump etc.

Your favourite scripting language

MORE THAN SECURITY TESTING

http://www.securing.pl/konsultacje

MORE THAN SECURITY TESTING

Thank you,

looking forward to contact!

jakub.kaluzny@securing.pl

INDUSTRIAL INSECURITY

Industrial insecurity

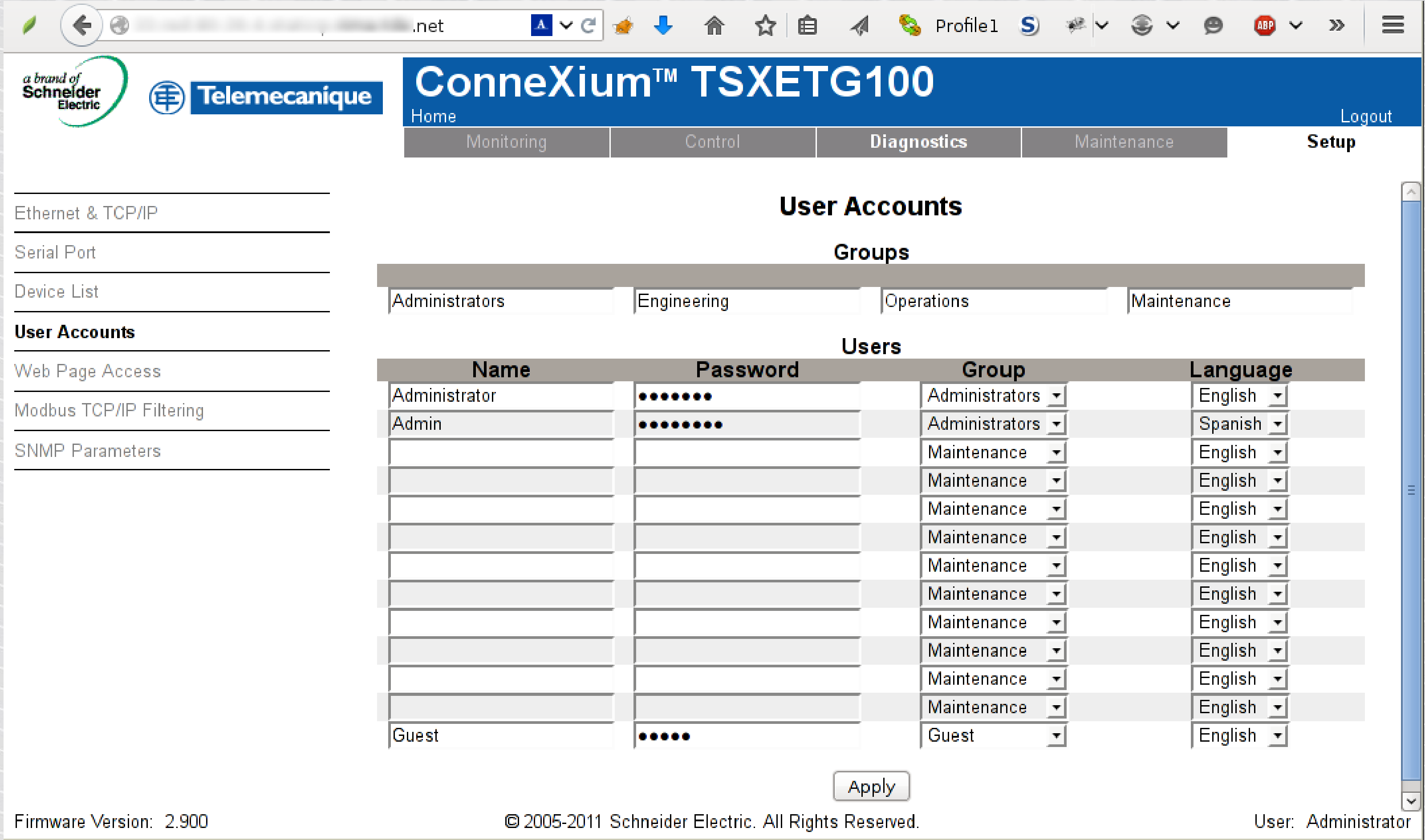
Thousands of interfaces publicly available.

Trivial to discover, already scanned & catalogued likewise cameras.

Modbus-TCP, Serial-TCP, default passwords or password-less web management interfaces...

I won’t reveal the links here ;)

Industrial insecurity – public interfaces



ConneXium™ TSXETG100

User Accounts

Groups

Administrators	Engineering	Operations	Maintenance

Users

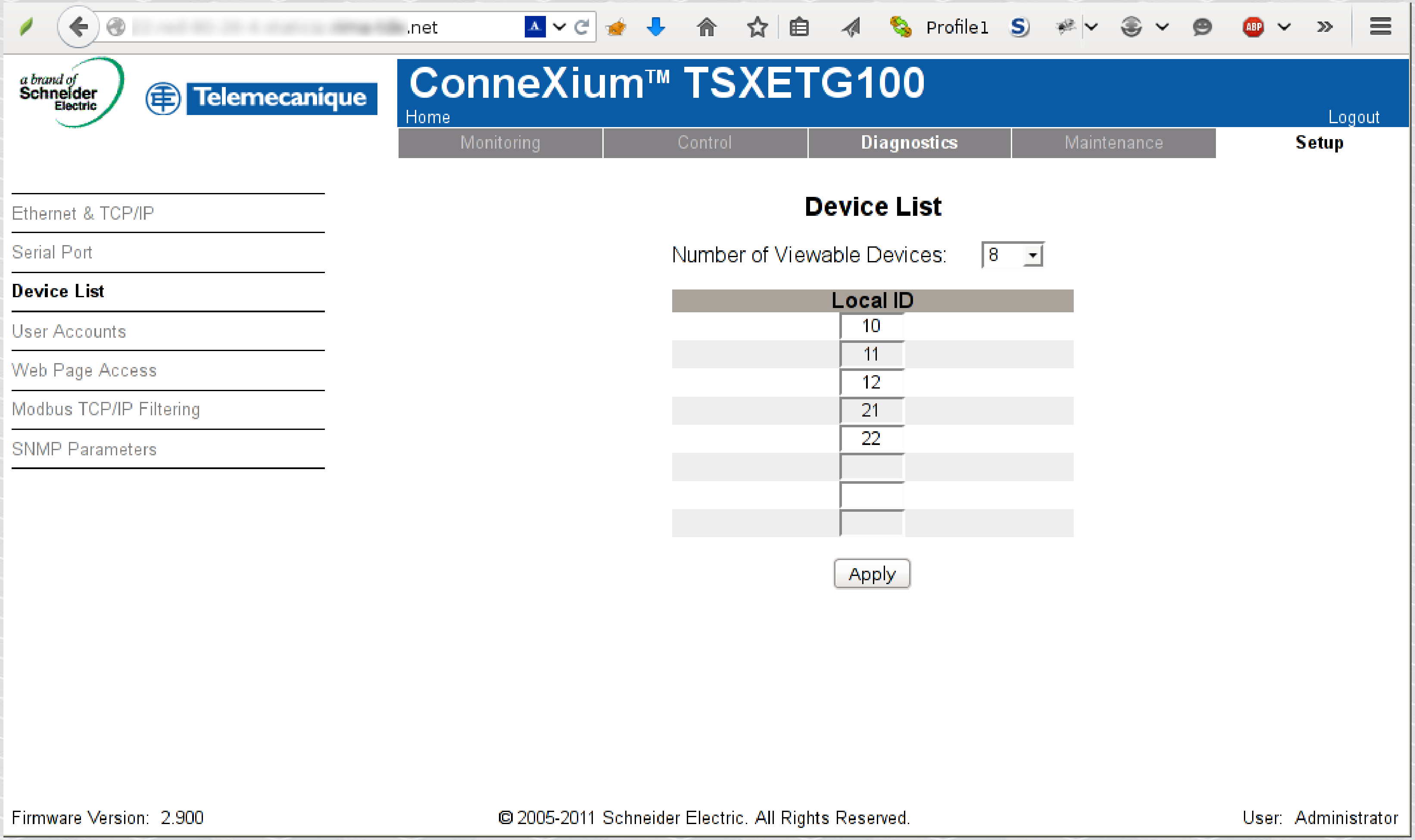
Name	Password	Group	Language
Administrator	●●●●●●●	Administrators	English
Admin	●●●●●●●●	Administrators	Spanish
		Maintenance	English
		Maintenance	English
		Maintenance	English
		Maintenance	English
		Maintenance	English
		Maintenance	English
		Maintenance	English
		Maintenance	English
		Maintenance	English
		Maintenance	English
Guest	●●●●●	Guest	English

Firmware Version: 2.900

© 2005-2011 Schneider Electric. All Rights Reserved.

User: Administrator

Industrial insecurity – public interfaces



[redacted].net

Profile1

a brand of Schneider Electric

Telemecanique

ConneXium™ TSXETG100

Home

Logout

Monitoring | Control | **Diagnostics** | Maintenance | **Setup**

- Ethernet & TCP/IP
- Serial Port
- **Device List**
- User Accounts
- Web Page Access
- Modbus TCP/IP Filtering
- SNMP Parameters

Device List

Number of Viewable Devices: 8

Local ID
10
11
12
21
22

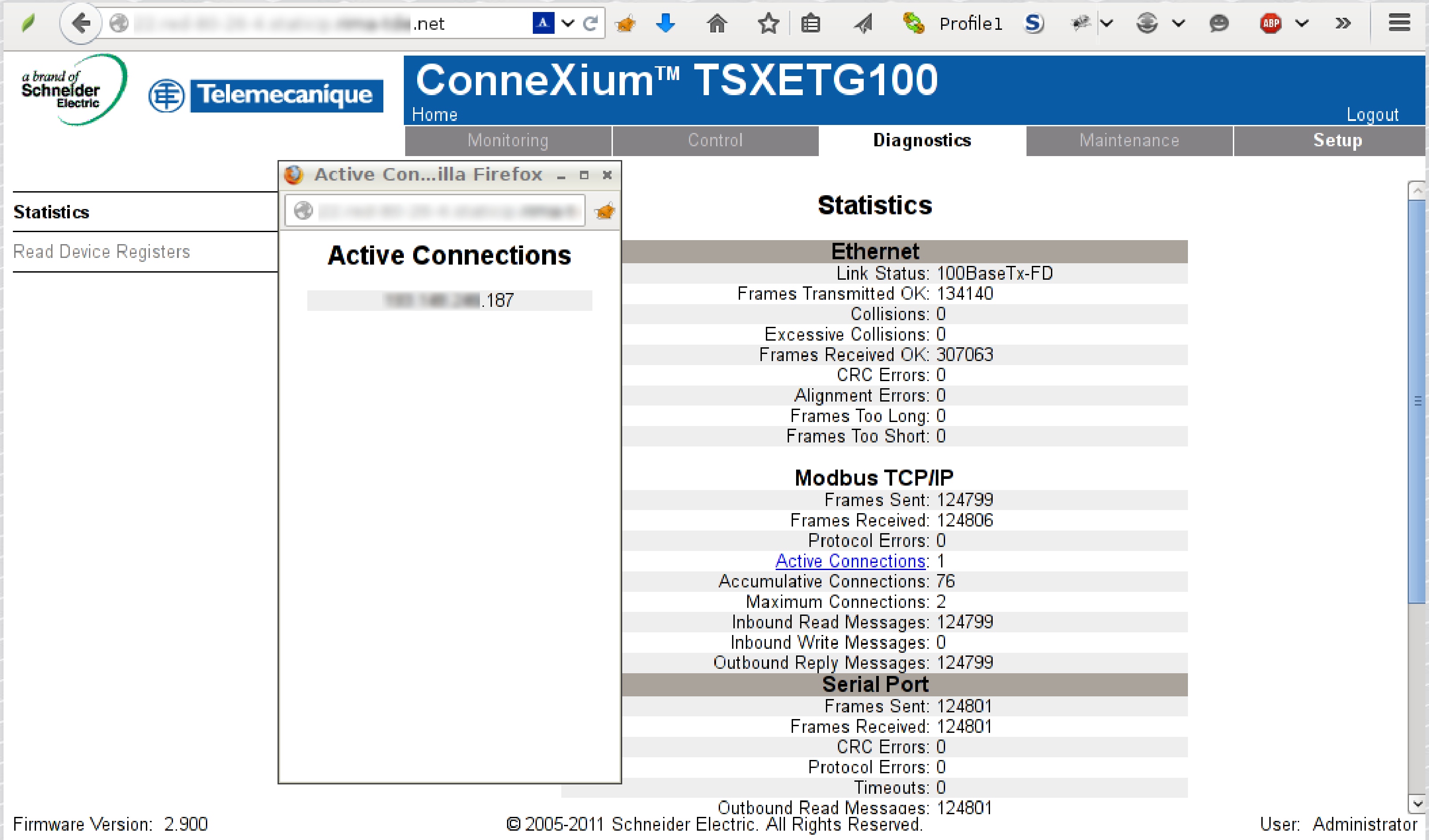
Apply

Firmware Version: 2.900

© 2005-2011 Schneider Electric. All Rights Reserved.

User: Administrator

Industrial insecurity – public interfaces



ConneXium™ TSXETG100

Active Connections

[redacted].187

Statistics

Ethernet

```
Link Status: 100BaseTx-FD
Frames Transmitted OK: 134140
Collisions: 0
Excessive Collisions: 0
Frames Received OK: 307063
CRC Errors: 0
Alignment Errors: 0
Frames Too Long: 0
Frames Too Short: 0
```

Modbus TCP/IP

```
Frames Sent: 124799
Frames Received: 124806
Protocol Errors: 0
Active Connections: 1
Accumulative Connections: 76
Maximum Connections: 2
Inbound Read Messages: 124799
Inbound Write Messages: 0
Outbound Reply Messages: 124799
```

Serial Port

```
Frames Sent: 124801
Frames Received: 124801
CRC Errors: 0
Protocol Errors: 0
Timeouts: 0
Outbound Read Messages: 124801
```

Firmware Version: 2.900

© 2005-2011 Schneider Electric. All Rights Reserved.

User: Administrator

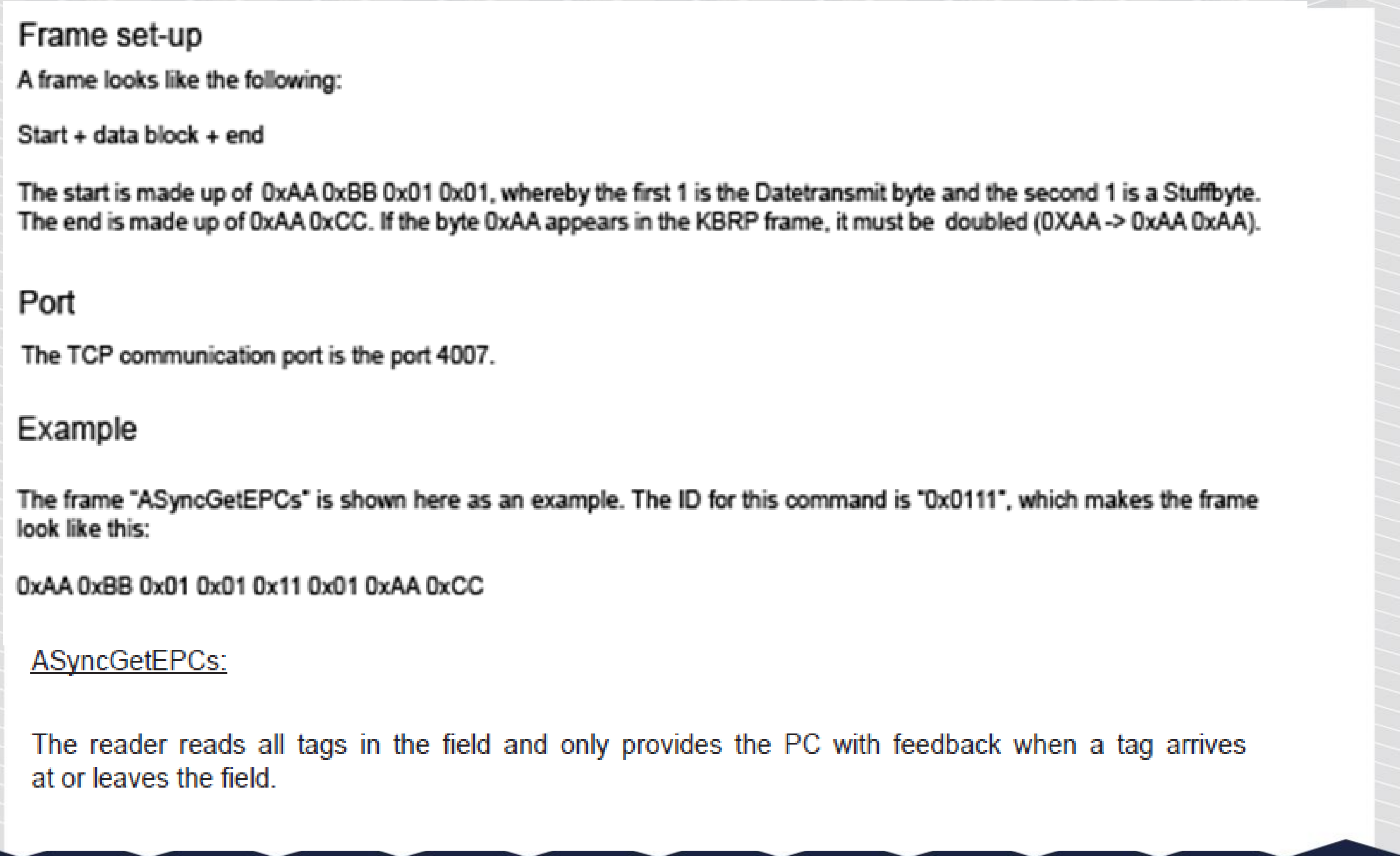
Industrial RFID reader

Read RFIDs mounted in privileged trucks to automatically open the gate.

Industrial RFID reader – port scan

```
PORT      STATE SERVICE    VERSION
23/tcp    open  telnet     Busybox telnetd
4007/tcp  open  pxc-splr?
4684/tcp  open  unknown
10001/tcp open  tcpwrapped
Service Info: Host: UHF-RFID-Dev
```

No need to hack - just RTFM



Frame set-up

A frame looks like the following:

Start + data block + end

The start is made up of 0xAA 0xBB 0x01 0x01, whereby the first 1 is the Datetransmit byte and the second 1 is a Stuffbyte. The end is made up of 0xAA 0xCC. If the byte 0xAA appears in the KBRP frame, it must be doubled (0XAA -> 0xAA 0xAA).

Port

The TCP communication port is the port 4007.

Example

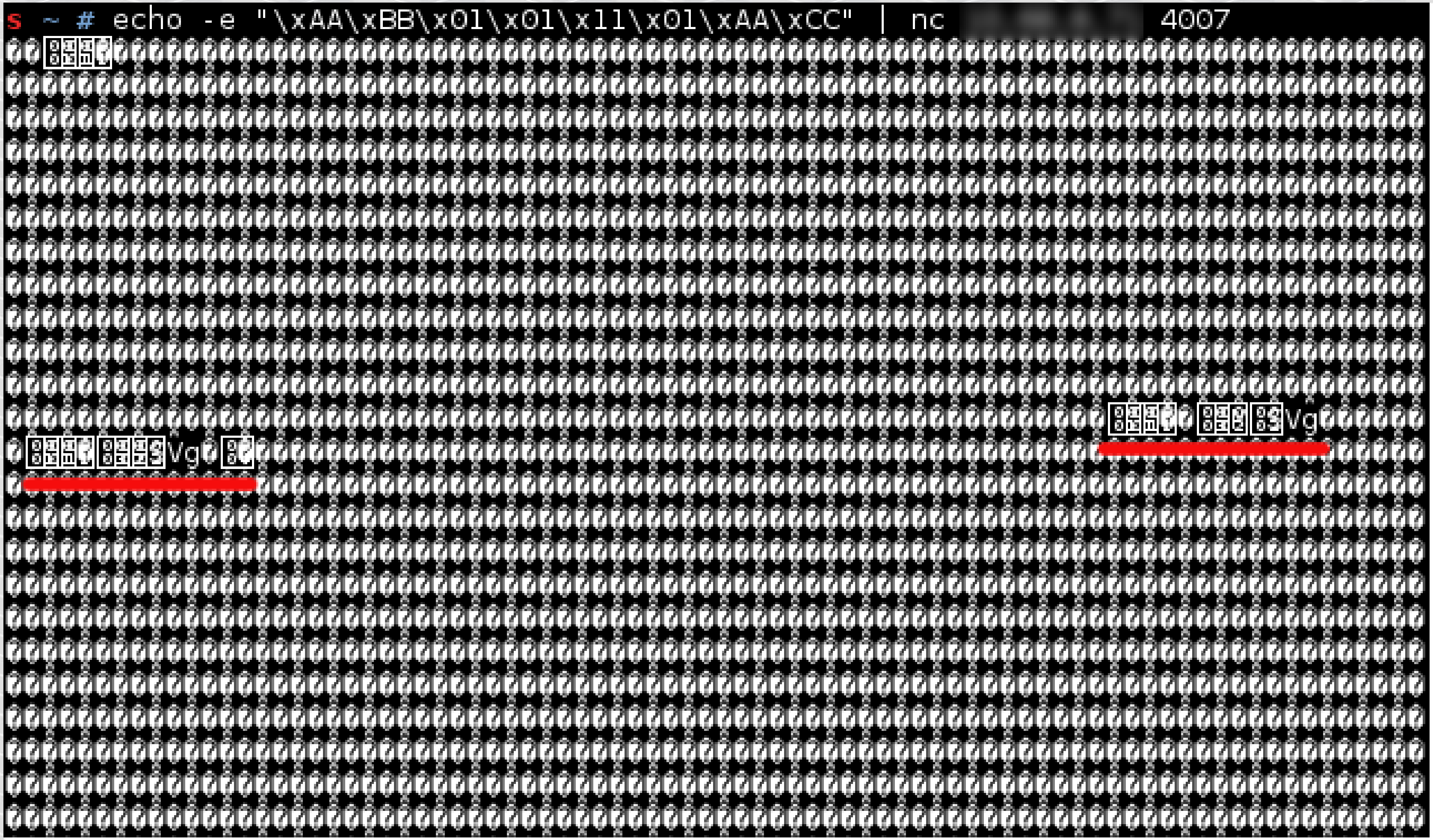
The frame "ASyncGetEPCs" is shown here as an example. The ID for this command is "0x0111", which makes the frame look like this:

0xAA 0xBB 0x01 0x01 0x11 0x01 0xAA 0xCC

ASyncGetEPCs:

The reader reads all tags in the field and only provides the PC with feedback when a tag arrives at or leaves the field.

Command-line „client”



```
s ~ # echo -e "\xAA\xBB\x01\x01\x11\x01\xAA\xCC" | nc [redacted] 4007
```

…and now we can clone the tag

```
$ echo -e "\xAA\xBB\x01\x01\x11\x01\xAA\xCC" | nc <IP> 4007 |
hexdump
0000000 bbaa 0101 8111 aa00 aacc 07bb aa00 aacc
0000010 07bb aa00 aacc 07bb aa00 aacc 07bb aa00
0000020 aacc 07bb aa00 aacc 07bb aa00 aacc 07bb
0000030 aa00 aacc 07bb aa00 aacc 07bb aa00 aacc
(...)
0000350 aacc 07bb aa00 aacc 07bb aa00 aacc 07bb
0000360 aa00 aacc 07bb aa00 aacc 07bb aa00 aacc
0000370 07bb aa00 aacc 01bb 1101 ffc1 0103 0247
0000380 1353 ed6b ccaa bbaa 0007 ccaa bbaa 0101
0000390 c111 0300 0001 5302 6b13 05ed aa00 aacc
(...)
```

Should we worry?

The incoming vehicles are also traditionally verified by security staff.

The device is available in restricted LAN only.

The tag can also be scanned from the truck itself.

BUT: you have to be aware of the technology shortcomings and not to alter the above conditions!

BLUETOOTH SMART

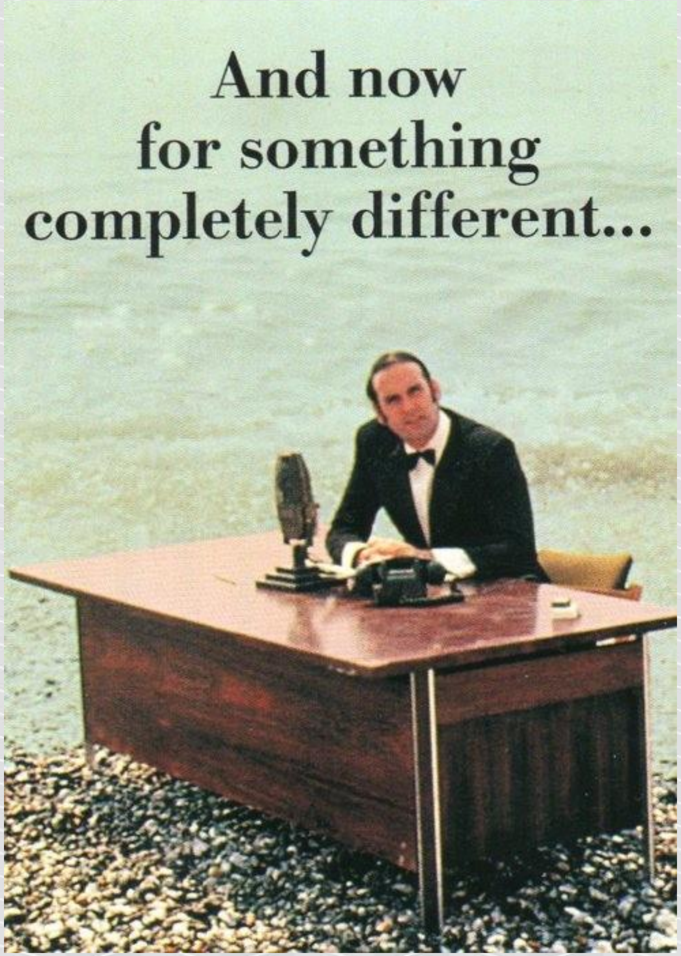
- AKA Bluetooth Low Energy, BLE, Bluetooth 4

Bluetooth Smart != Bluetooth 3

Completely different stack – from RF to upper layers.

Designed from the ground-up for low energy usage.

Network topology
a) Broadcaster + Observer
b) Master + Peripheral



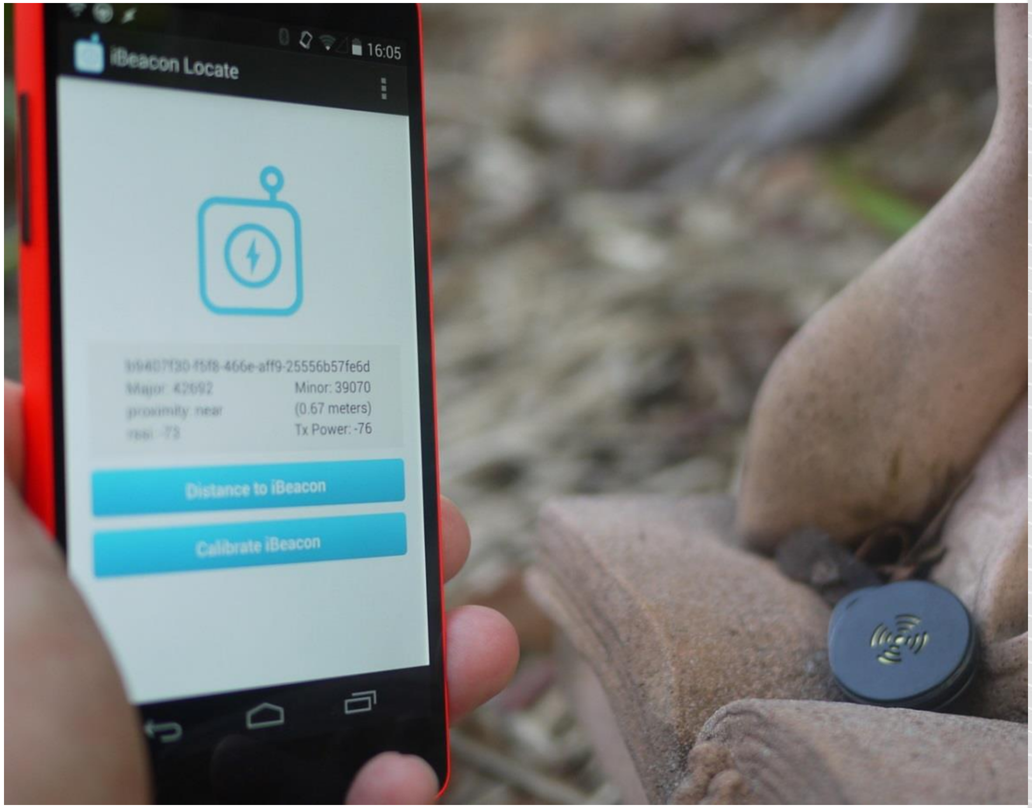
Broadcast - beacon

UUID (vendor)
2F234454-CF6D-4A0F-ADF2-F4911BA9FFA6

Major (group)
45044

Minor (individual)
5

Tx Power
-59

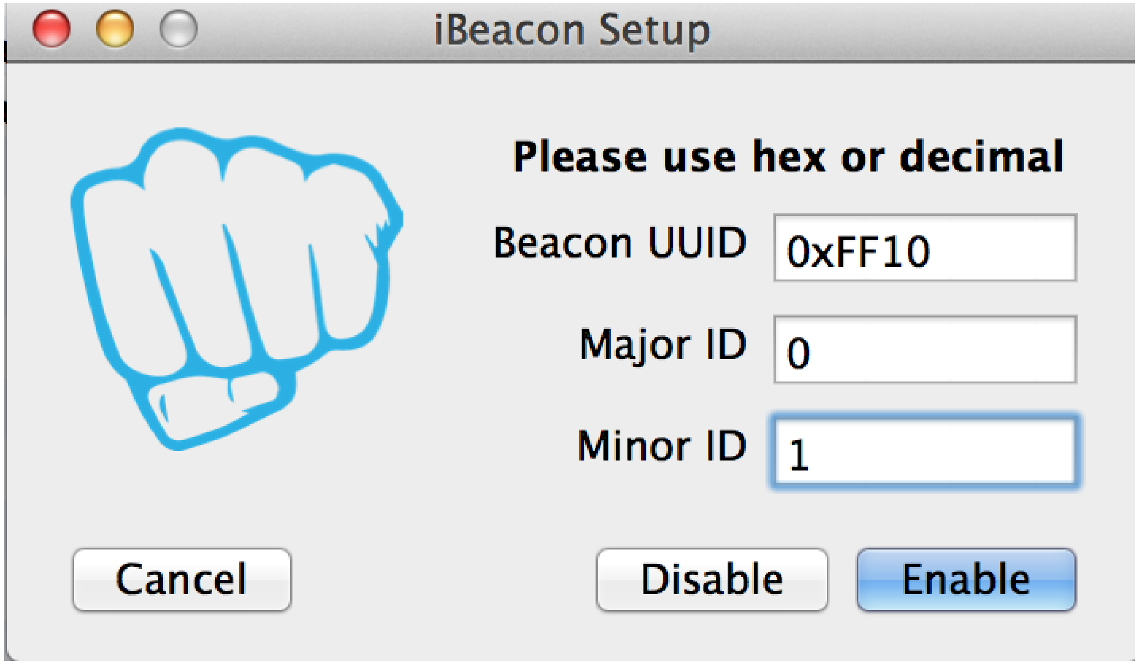


https://www.flickr.com/photos/jnxyz/13570855743

The mobile app can measure precise distance to specified beacon.

Beacons – emulation #1: LightBlue

Available for iPhone, iPad, Mac



iBeacon Setup

Please use hex or decimal

Beacon UUID: 0xFF10

Major ID: 0

Minor ID: 1

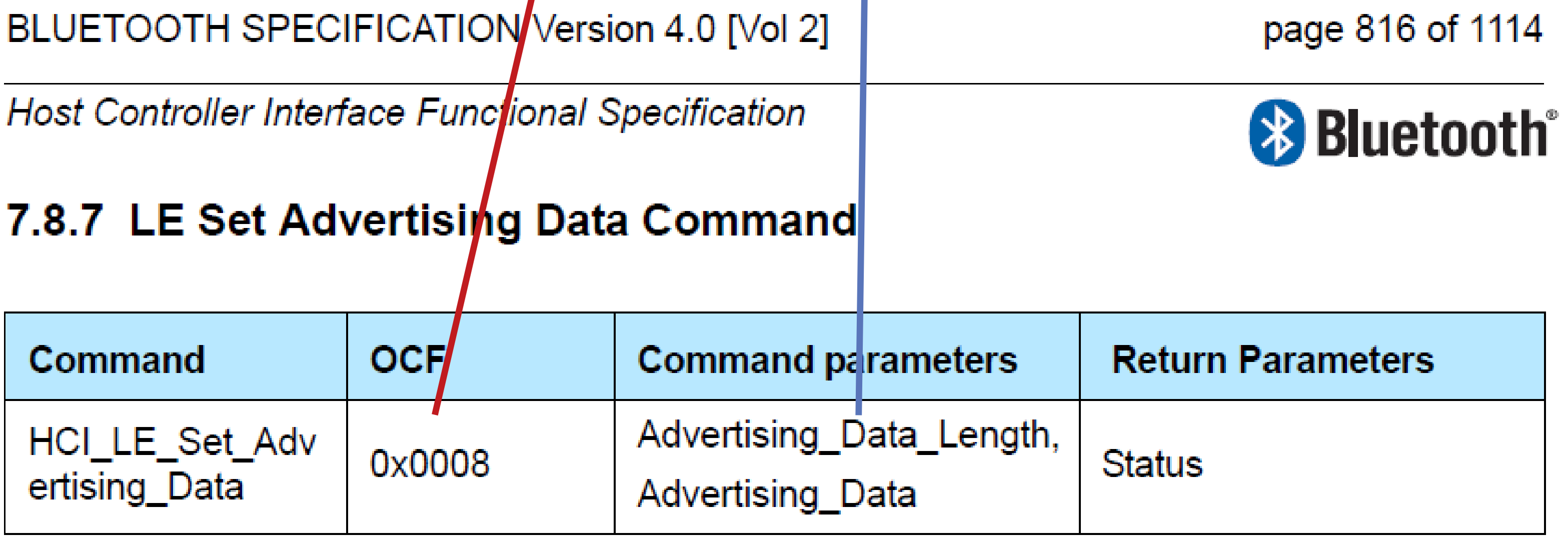
https://itunes.apple.com/us/app/lightblue-bluetooth-low-energy/id557428110

Beacons – emulation #2: Bluez

```
# hcitool cmd 0x08 0x0008 1E 02 01 1A 1A FF 4C 00 02
15 84 2A F9 C4 08 F5 11 E3 92 82 F2 3C 91 AE C0 5E FD
E8 AF C8 C5 00
```

Beacons – emulation #2: Bluez

```
# hcitool cmd 0x08 0x0008 1E 02 01 1A 1A FF 4C 00 02
15 84 2A F9 C4 08 F5 11 E3 92 82 F2 3C 91 AE C0 5E FD
E8 AF C8 C5 00
```



BLUETOOTH SPECIFICATION Version 4.0 [Vol 2] page 816 of 1114

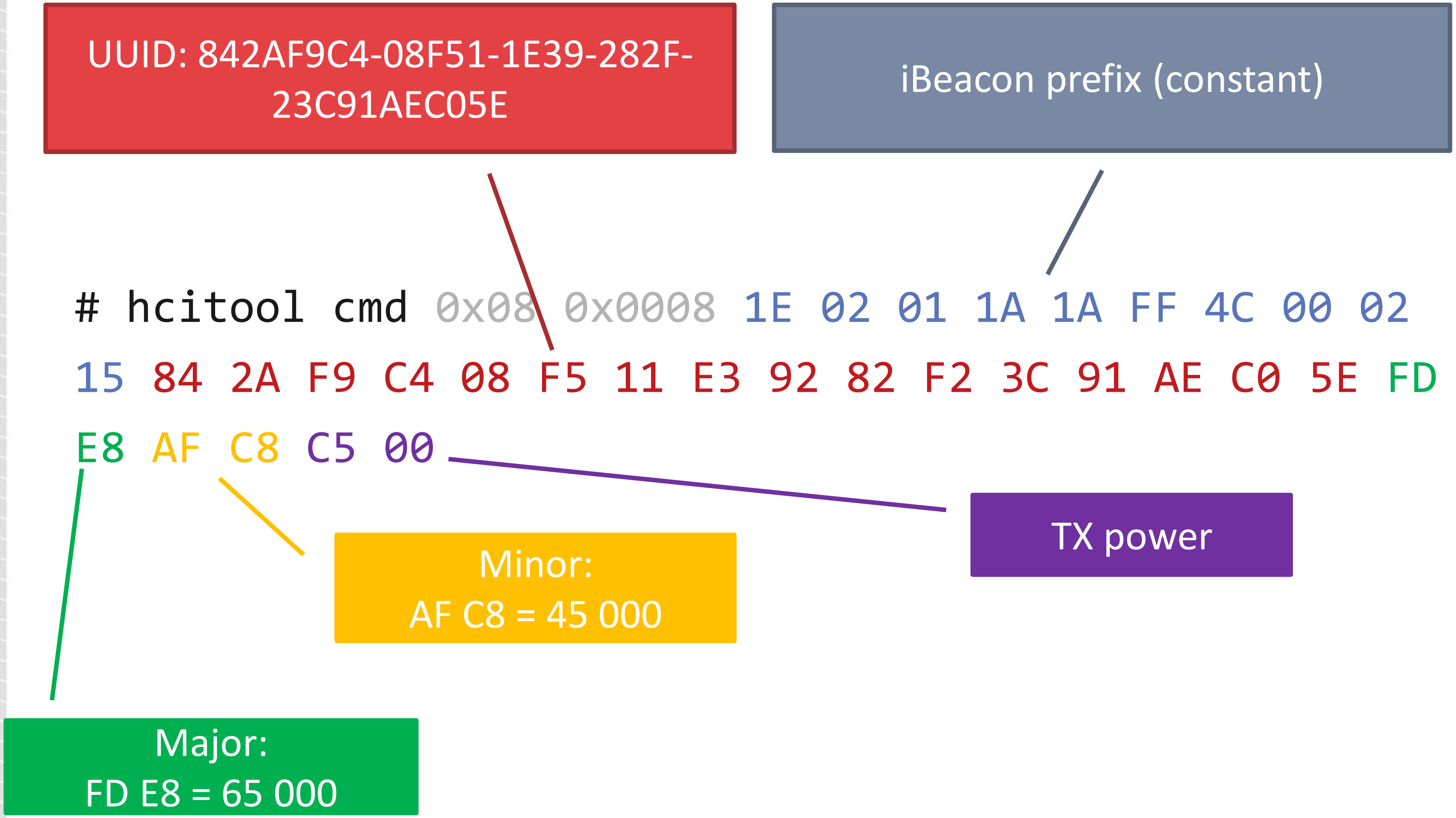
Host Controller Interface Functional Specification

Bluetooth®

7.8.7 LE Set Advertising Data Command

Command	OCF	Command parameters	Return Parameters
HCI_LE_Set_Advertising_Data	0x0008	Advertising_Data_Length, Advertising_Data	Status

iBeacon data broadcast



UUID: 842AF9C4-08F51-1E39-282F-23C91AEC05E

iBeacon prefix (constant)

```
# hcitool cmd 0x08 0x0008 1E 02 01 1A 1A FF 4C 00 02
15 84 2A F9 C4 08 F5 11 E3 92 82 F2 3C 91 AE C0 5E FD
E8 AF C8 C5 00
```

TX power

Minor:
AF C8 = 45 000

Major:
FD E8 = 65 000

Beacons – some example usage scenarios

Additional info on products based on precise location.

Rewards for visiting places.

Indoor guide, help to navigate the blind etc.

Your home or toys can automatically react to you.

Be warned that your bike or car is no longer in the garage.

Beacons – additional info based on location

Abuse?

Beacons – the navigating usage scenario

CRUISE CONTROL FOR PEDESTRIANS:
Controlling Walking Direction using Electrical Muscle Stimulation

Max Pfeiffer, Tim Dünte, Stefan Schneegass, Florian Alt, Michael Rohs

Abuse?

OTHER BLE DEVICES

Beacons are just the beginning…

How to make your own BLE device?

1. Buy SDK+devices from selected vendor (Nordic, TI...)
2. Import ready-to-use sample code.
3. Add your bright usage scenario (and sometimes a bit of hacking).
4. Create convincing bootstrap webpage + videos.
5. **Run successful Kickstarter campaign.**
6. Profit!

Beacons are just the beginning...

Electric plugs, lightbulbs, locks, kettles, sensors, wallets, socks, pans, jars, toothbrushes, bags, plates, dildos, sitting pads, measuring your farts devices, calorie-counting mugs...

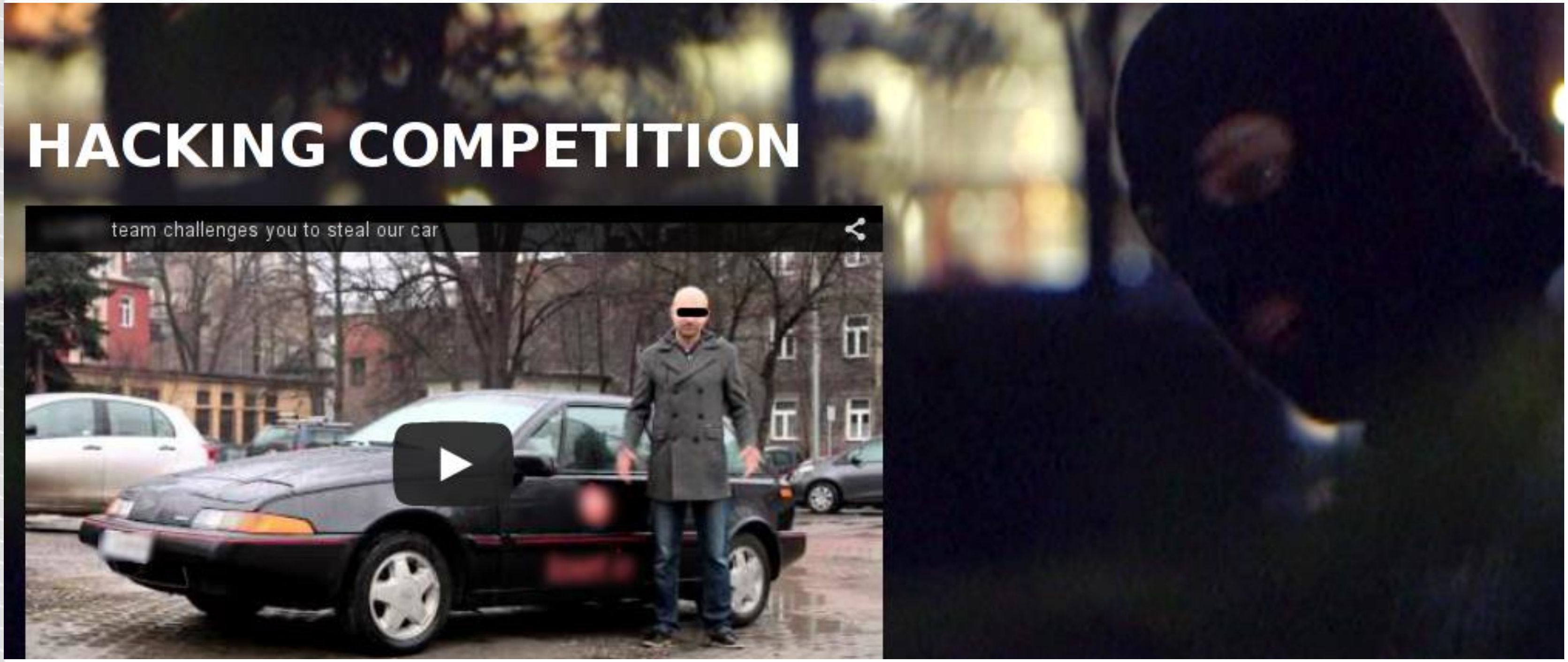
„It was just a dumb thing. Then we put a chip in it. Now it's a smart thing.”

(weputachipinit.tumblr.com)

Crowdfunding: a new kind of celebrity. Too often ridiculous meets big money.

www.myvessyl.com

Other BLE devices

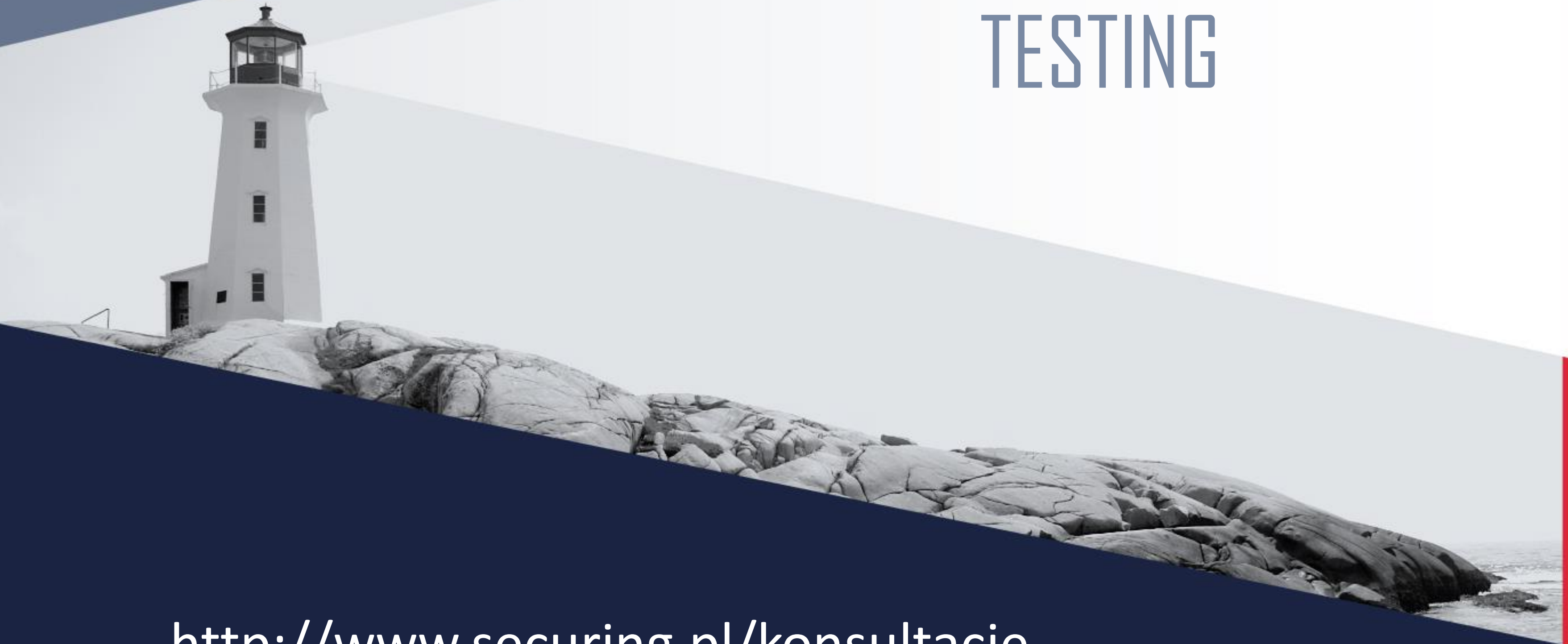


They have been assured the communication is unbreakable because they use AES.

I showed an intruder may approach the unsuspecting victim’s phone once (even with autounlock feature off), to be able to get full control over the car for consecutive times without consent of the victim.

securing

MORE THAN SECURITY TESTING



http://www.securing.pl/konsultacje

securing

MORE THAN SECURITY TESTING

Thank you,

looking forward to contact!

jakub.kaluzny@securing.pl