11 – Security

What is computer security?

As computers are more and more interconnected, the problem of keeping them *secure* becomes more and more complex

… but what does *security* mean, in this field?

- The C.I.A. Paradigm
  - Confidentiality (my data are unknown to others, if they are not authorized to access them)
  - Integrity (data remain the same if I don't change them)
  - Availability (if I am authorized to access some data, they are accessible in a convenient format and in a reasonable time)

What is computer security?

- The main concepts of the C.I.A. paradigm are strongly connected with the following ones:
  - Identification (who do you say you are?)
  - Authentication (how do I know it is really you?)
  - Authorization (what are you allowed to do?)
  - Accountability (who did what?)

… what does it mean?

- How can we translate these concept into real life terms?
  - Confidentiality
    - my email account is safe, nobody else can read it
  - Integrity
    - my blog is safe, no unauthorized people can delete my posts
  - Availability
    - I can always connect to the services I rely on for my work or my leisure time

Are we secure?

- The purpose of this lesson is to show you how much we are (not) secure
  - … of course, I will depict things in a slightly negative way, however…

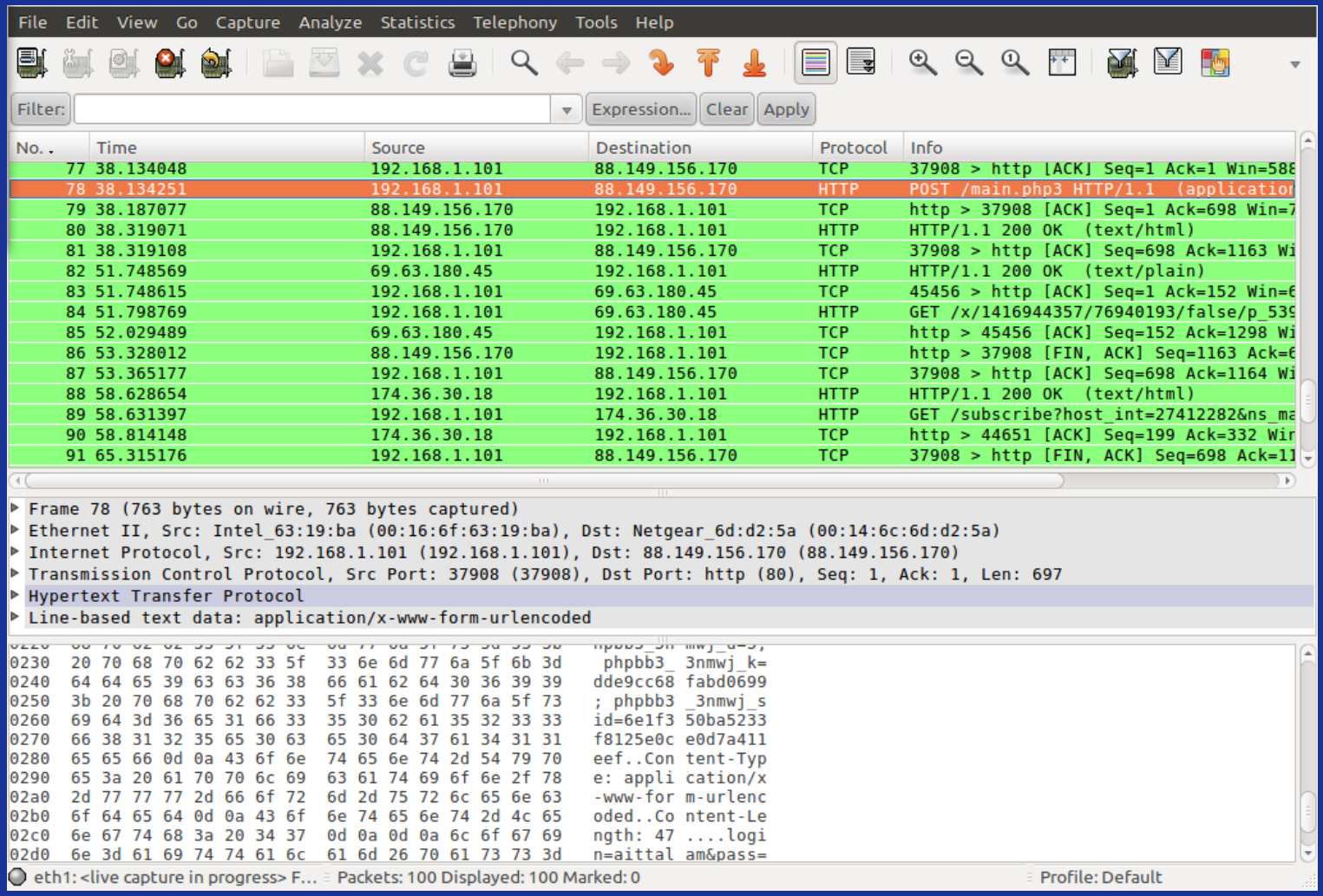
Example 1: email spoofing

- Play with your own SMTP email server
- … and learn to understand who actually sent you an email (hint: do not trust the “From” field ;-))

```
MAIL FROM: whoever.youwant@wherever.youlike
250 2.1.0 Ok
RCPT TO: blablah@gmail.com
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
Subject: this is a spoofed message

Hi there,

I am not who you think I am
:-)

.
250 2.0.0 Ok: queued as D3E7840131
```

Example 2: sniffing data

- Always choose encrypted channels when possible, otherwise...



Example 3: SQL injection

- A rather old, but still valid technique (see here and here)
- Avoid mistakes such as:

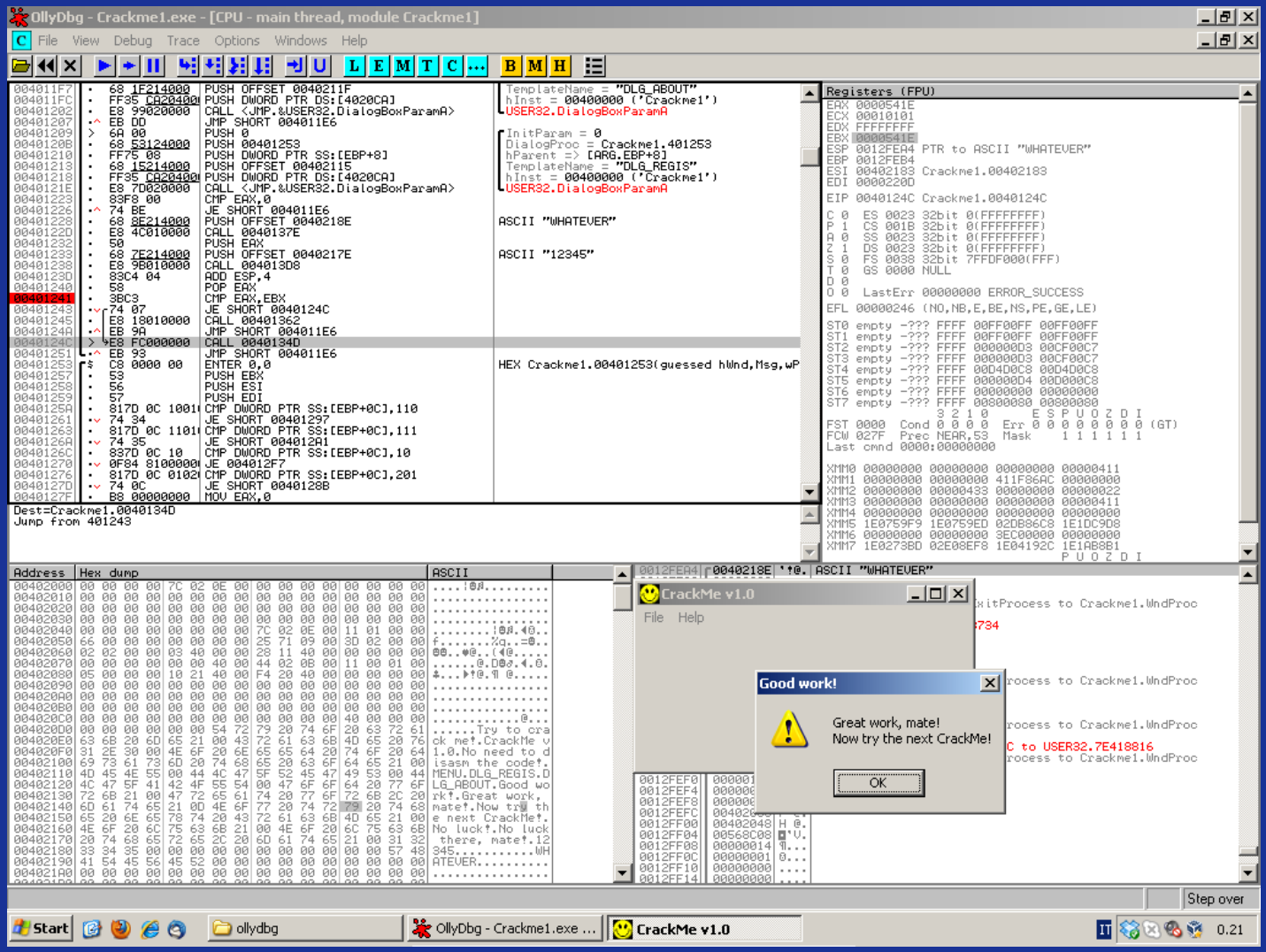
```php
// get the login and pass and check if they are correct
$login  = $_REQUEST['login'];
$pass   = $_REQUEST['pass'];
$query  = "select * from user where login='$login' and pass='$pass'";
$result = mysql_query($query);
```

… as the result might be:

The query is: select * from user where login='test01' and pass='' or '1'
Welcome, user id 1

Example 4: code reversing

- Do not trust your code to be secure just because it is compiled.



Are we secure?

- … from the examples we have seen we can learn some important lessons:
  - there is no “out of the box” secure solution, as security depends on many different factors
  - “security by obscurity” is NOT security
  - most of the times, security is just related to knowing (well!) how things work. The more you know the more chances you have to make your system secure... or at least know what you risk!

References

- Tools used for this lesson:
  - Wireshark: http://www.wireshark.org/
  - OllyDBG: http://www.ollydbg.de/