Secure Programming via Libraries (T3)
ECI 2011 - Day 3

Alejandro Russo

Chalmers University of Technology

Exercise 1 Implicit flows and taint analysis

Taint analysis usually tends to ignore implicit flows. In that manner, it is possible to circumvent the analysis if the attacker has the chance to write the code.

1. In this exercise, you should implement the function `implicit` that converts a tainted character into an untainted one by implicit flows. We only consider characters that are between the ASCII code `0` and `255`. Your code should pass the following tests.

```
>>> tc = taint('a')      # This is a taint character
>>> tc
'a'
>>> tainted(tc)
True
>>> uc = implicit(tc)    # tc becomes untainted
>>> uc
'a'
>>> tainted(uc)
False
```

2. Now that you know how to untainted a character, without applying sanitization functions, write the function `convert` that untaints a whole string. Assuming that you have the following code

```
from taintmode import *

@ssink(OSI)
def execute(cmd):
    # Instead of printing, here we run the command cmd in the shell
    print "Executing:", cmd


attack = taint('rm -r *')

execute(attack)
```

Then, the library should report a potential attack. In contrast, if you now write

```
from taintmode import *

@ssink(OSI)
def execute(cmd):
    # Instead of printing, here we run the command cmd in the shell
    print "Executing:", cmd


attack = taint('rm -r *')

execute(convert(attack))
```

The taint analysis does not detect the attack since tainted data has been converted into untainted by implicit flows.

Exercise 2 Completness of taint analysis

The previous exercise shows that taint analysis could be usually not sound, i.e., tainted data might affect security critical operations and the analysis could miss it. Taint analysis, like most of other analysis, is not complete either, i.e., there exists programs which are secure but the analysis raises an alarm.

1. Can you write a program that it is secure, i.e., the content of tainted data does not affect the secure critical operations but the analysis raises an alarm?