2 Assignment 2 (10 Points)

Hinweis:	Abgabe in $\{2, 3, 4\}$-er Gruppen.
Abgabe:	**06.05.2017, 23.59**
Email:	Betreff "[Compsec] Ex2"
	(bitte nur .pdf oder .txt, kein .doc, .jpeg, etc)
	Source code: bitte inkl. `signify` Signatur

Exercise 3 (Case study μ-`shout` (I): a small echo server (**6 Points**))**.**
Write a **C**-Programm, that

1. listens for (unencrypted) TCP connections on some port,
2. reads a message from this connection and writes it back (echo server),
3. and on each event (connect and message), writes an entry in a log file at `/var/log/ushoutd.log`.

Decide for yourself which entries you want to include in the log file [6] but make sure that only the root user is allowed to read and write the file.

Write a `Makefile` to compile your program and research which compiler flags will help you write the most secure code. Your program should compile without errors and warnings and run on a i386 OpenBSD system.
Hint: You may want to use netcat `nc(1)` to test your server (you don't have to program a client by yourself yet).
Note: You probably need to run your program as `root`.

Exercise 4 (Real-world de-anonymization **2+2 Points**)**.**
Read `https://33bits.org/2008/11/12/57/` and download the current "Loan Data" history from `https://www.lendingclub.com/info/statistics.action`.

1. The blog post from above is quite old. Which steps can you identify that Lending Club has taken to "anonymize" the data?
2. Find the identity (name or social-media account or homepage, etc) of one lender and describe your process (note that some lenders are easier to identify than others).

Exercise 5 (Keeping your systems secure (**Bonus: 1 Points**))**.**
Are there any new vulnerabilities for your Debian or OpenBSD system since last week (29.04.2016 at 23.59)? If so: state one, name the programming mistake, decide if you are affected or not, and report if there are any known work-arounds or patches.

[6] You do not have to "anonymize" the entries.