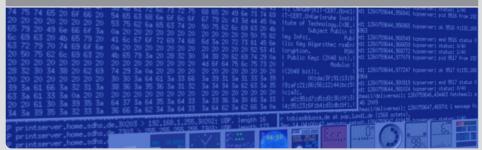


## SCC4 Debriefing — Forensics

Tobias Dussa • GridKa School 2010

#### COMPUTER EMERGENCY RESPONSE TEAM



#### Introduction & Overview



## This talk addresses the following questions:

- Why forensic analysis?
- Where and how to gather evidence?
- How to analyze evidence data?

#### It does not address:

- How to contain damage?
- What to communicate when to whom?
- How to recover from an incident?



## What Is Forensic Analysis Good For?



To assess and answer several important questions about an incident:

- Where did the attacker come from?
- How was access gained?
- What damage was done?
- What other machines were affected?
- ... and many more related questions.

## Case Study: SSC4 Situation



- Known facts for Security Service Challenge 4:
  - IP addresses 192.108.46.248 and 195.140.243.2 are evil.
  - (End of list!)
- Unknown: Everything else, particularly
  - Are the bad IPs involved with our systems?
  - If so, how?
  - And what happened, if anything?

## **Data Sources for Forensic Investigations**



- Broad classes of data sources:
  - 1. Highly volatile (e.g., CPU registers),
  - 2. Volatile (e.g., RAM),
  - 3. Static (e.g., hard drives), and
  - 4. Highly static (e.g., archive tapes).
- More volatile evidence must be gathered and preserved first, if possible.
- Obviously, not all classes available or applicable in every instance.

## **Back to SSC4: Initial Investigation**



- First step: Find out whether the IPs in question have shown up at our site.
- Sifting through the appropriate logs yields a machine connected with the suspect IPs (boring).
- Watch out for timestamps (time zone used)!

## So We Have a Suspect ...



- ... or at least a suspect machine. Now what to do?
- 1. Identify the suspect processes.
- 2. Gather all volatile evidence.
- 3. Gather less volatile evidence.
- 4. Work out what happened.

#### Who's Who



## How to find suspect processes?

- Why, with ps, of course.
- ... and with netstat.
- ... and with lsof.
- Watch out for
  - processes with weird process names,
  - processes that belong to the init process,
  - processes that hold suspect network sockets or connections.



#### **Back to SCC4**



## Process was quickly identified (no stealth measures).

Process belonged to a job submitted with a user certificate with the DN

```
/O=XXX/O=XXX/O=XXX/CN=XXX (SSC4).
```

Next step: collect all available information, using for instance the following tools:

```
arp, gdb, ip, ipcs, last, lastlog, lsof,
mount, netstat, ps, w, who
```

## Gimme, Gimme, Gimme



## Volatile process data to be secured includes:

- The executable binary of the process being executed,
- the core dump of the process,
- environment variables and settings, such as
  - the current working directory,
  - shared memory regions,
  - limits, and
  - open file handles.

#### **Gimme Even More**



#### And also save some static data:

- All related log files:
  - Machine,
  - gateways,
  - servers,
  - NATs,
- the user's home directory (watch out for privacy though!),
- actually, if possible, the entire file system.

## After Compiling, Interpretation!



## Take a close look at the collected data. Some pointers:

- Inspect suspect executables (with, for example, strings, hexdump, gdb, rec, or more sophisticated disassemblers like IDAPro if available).
- Look at core dumps (using gdb).
- Grep through log files and the like.
- Check files' MD5 sums against the known-good list (you have one, right?).
- Perform further filesystem analysis, for instance with autopsy or rkhunter.



#### **SSC4 Revisited**



- Running the binary through strings reveals some fishy strings in the binary:
  - JOIN, NICK, PONG, PRIVMSG, USER
- Disassembling yields information about:
  - Communication and
  - other activity.
- Inspecting the core dump gives actual ID strings used in communication.

#### **SSC4 Encore**



## After detailed analysis, the following facts were known:

- Binary was an IRC bot (communication endpoints and parameters known),
- (tried to) install
  - at job and
  - cron job

to become persistent, and

- (tried to) transfer /etc/passwd out to drop site, but
- no root exploit used and no root kit installed.



#### **Common Pitfalls**



## Things to watch out for when doing forensics:

- Modifying evidence while collecting (e.g., file access times).
- Dropping volatile evidence (e.g., memory content).
- Failing to document actions properly (timestamps!).

## **Good Approaches**



## Common sense and good practices:

- Strictly separate evidence acquisition and evaluation.
- Gather evidence, then produce a working copy of the evidence locker, then work on the working copy only.
- Go out of your way to ensure you work in read-only mode whenever possible, even on the working copy.
- And, most importantly, if you are unsure what to do, talk to somebody who has a better chance of knowing (i. e., EGI CSIRT).

## Th-Th-Th-Th-Th-... That's All, Folks!



# Any questions?

## Th-Th-Th-Th-Th-... That's All, Folks!



## Thank you for your attention!