

REMINDER: MANDATORY FINAL "QUIZ" IS NEXT WEEK, DECEMBER 6TH!

What types of jobs would fall under "Production Operations" or "Production Engineering"?

- System Administrator, System Engineer
- Network Administrator, Network Engineer
- Network Operations Center (NOC) employees (alert response, triage troubleshooting)
- (Site) Reliability Engineer (SRE) (often a lot of engineering involved - code debugging, etc)
- Other "Dev-Ops" or "Ops-Eng" labeled roles
 - Generally people with strong programming/dev backgrounds that also have great operational troubleshooting (often black-box) skills ("Reliability Engineering")
 - Operations-related development (monitoring, tools, etc)
- Database Administrator/Engineer (DBA)

Something that operations will usually have to endure are the "3am wakeup calls", where "something is wrong, but we don't know what". That may sound like a crappy situation (woken up in the middle of the night), but when these problems are (usually) new ones with something new to troubleshoot or debug, it can actually be pretty exciting! This is especially true when these issues are ones that are affecting (potentially) millions of people (even if it's thousands, it's still important!).

Work with engineering to ensure overall architecture is scalable. Not only system/network-level, but also design (what algorithms are used, how efficiently things are done -- especially considering the hardware involved, etc).

One of the worst things is to have hardware that does not "match" the software that is going to be deployed on it. Operations needs to work closely with Engineering to ensure everyone is on the same page -- which is why "dev ops" type roles are often very important. I've had way too many "engineers" simply not even really care or think about the hardware they are using -- which will always end with something going wrong... (cf. previous Facebook photos example... and very old feed example)

Making sure software behaves within expected parameters.

Example from Loudcloud (before I had implemented my DNS tool, mind you! ;))...Got a call late in the evening that "DNS stopped working". After some troubleshooting, found the problem: zones were not transferring from the master - claiming the SOA (Start of Authority) serial number was older than current. Looking at the serial numbers (which are used by the master and slave domain name servers to determine if a zone needs to be refreshed or not), it was evident the existing DNS script had some bad logic in it.

DNS SOA serial numbers are in the format YYYYMMDDRR, which (should) ensure ever-increasing numbers, and fit within 32 bits:

```
YYYY = full, 4-digit year
MM   = 0-padded month
DD   = 0-padded day
RR   = 0-padded revision for the day
```

What was happening was that the existing DNS update script was just blindly incrementing the DNS serial number's "Revision" and started appending a 3-digit number (we somehow had over 100 revisions for that day). Since the SOA Serial is an unsigned 32-bit integer, the 3-digit revision number was causing overflow. Problem solved, but could have been averted if the original script had had better logic in it (which I employed when I later created my DNS tool ;)).

Books: Both are by "Martin L. Abbot" and "Michael T. Fisher"

Scalability Rules – 50 principles for scaling web sites

The Art of Scalability – Scalable Web Architecture, Processes and Organizations for the Modern Enterprise