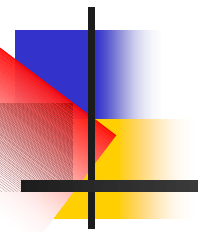
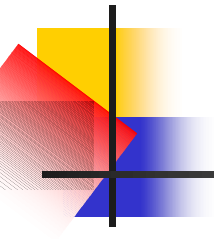


AT&T – Ausfall des Telefonnetzes in den USA

von

Jörg Sesterhenn



Gliederung

- Was ist passiert?
- Was waren die Ursachen?
- Mit welchen Maßnahmen hätte der Fehler verhindert werden können?
- Was können wir daraus lernen?
- Welche anderen schwerwiegenden Software-Fehler hat es in diesem Bereich gegeben?

Was ist passiert?

- AT&T-Telefonsystem:
  - Zentralstelle in New Jersey
  - 114 vernetzte regionale Schaltzentralen
- 15. Januar 1990: Fehlfunktion in einer Schaltzentrale in Manhattan führt zur Kettenreaktion
- 70 Mio. von 138 Mio. Ferngesprächen innerhalb USA konnten 9 Stunden lang nicht vermittelt werden

Was ist passiert?

Die Schaltzentrale in New York setzte sich nach einer Fehlfunktion in den RESET-Modus:

- Ausfall-Meldung an alle anderen Zentralen
- Neubesetzung interner Tabellen (Reset)
- OK-Meldung an alle anderen Zentralen
- Weiterleiten neuer Ferngespräche
- Alle Zentralen mussten daraufhin ihre Tabellen ändern

Was ist passiert?

- Bei 3 Schaltzentralen kamen kurz nach der OK-Meldung neue Gespräche an
- Verarbeitung der Meldung und der neuen Gespräche führte zum Rechnerausfall
- Im Schneeball-System wurden 9 Stunden lang alle Zentralen lahm gelegt

Was ist passiert?

Notlösung:

- Kurzzeitige Reduzierung der Nachrichtenlast
- alte stabile Version wurde wieder eingesetzt

Weitreichende Konsequenzen:

- Verdacht auf einen Hackerangriff

➔Novellierung der Gesetze zur Computerkriminalität

Was ist passiert?

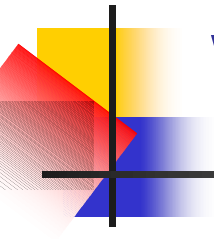
Schaden:

- $ 75 Millionen bei AT&T
- Mehrere $ 100 Millionen bei den Kunden
- Firmen-Image
- Vertrauensbruch
  - zu AT&T
  - zur Technik i. a.

Was waren die Ursachen?

- „...calling volume was not unusual...“
- „...a series of events that had never occurred before...“

AT&T's official report



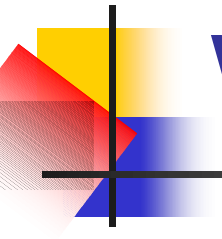
Was waren die Ursachen?

Software Update

- break-Befehl wurde falsch eingesetzt
- Update wurde direkt im größten Teil des Systems durchgeführt

Was waren die Ursachen?

```
switch expression {
        ...
        case (value):
            if (logical) {
                <statements>
                break;
            } else {
                <statements>
            }
            <statements>
        ...
    }
```



Was waren die Ursachen?

Fehler beim Testen & Validieren:

- Software war nicht hinreichend getestet
- latenter Fehler: existierte seit Programm-Optimierung 4 Wochen zuvor

Was waren die Ursachen?

Technologische Revolution –

Systeme stärker automatisiert

- Systeme komplexer und undurchsichtiger
- direkter Eingriff nicht mehr möglich
- automatische Sicherheitsroutinen gegen Pannenszenarien

Wie hätte der Fehler verhindert werden können?

Internationale Vermittlungsstelle
New York, 1936

Vermittlungszeit
ca. 2 Minuten

Wie hätte der Fehler verhindert werden können?

AT&T Techniker ersetzt eine Platine in einem 4ESS Switch
Los Angeles, 1980

Vermittlungs-zeit ca. 2 Sekunden

Wie hätte der Fehler verhindert werden können?

- dezentrales Telefonnetzwerk bietet bestmöglichen Schutz vor Totalausfall
- 1929-1980 hierarchisches Netzwerk
- seit 1980 dynamisches Routing (4ESS)
- 1990
  - ca. 40 Mio. Anrufe / Tag
  - 114 Schaltzentralen

Wie hätte der Fehler verhindert werden können?

- Sicherheits-Update wurde gleichzeitig auf 80 Schaltzentralen eingespielt
- Update in mehreren Schritten hätte Fehlerausmaß verringern können

Wie hätte der Fehler verhindert werden können?

Verstärktes Testen und Validieren hätte den Fehler wahrscheinlich aufgedeckt:

Firma „Software Research“ behauptet:

>>The error, [...] would have been revealed with attainment of complete C1 coverage <<

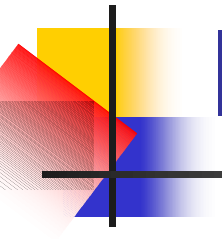
Wie hätte der Fehler verhindert werden können?

- Bei sicherheitskritischen Systemen ist Verifikation notwendig
- Verifikation von Teilaspekten ist aufwendig, aber möglich
  - hätte im Fall der fehlerhaften Break-Anweisung zum Erfolg geführt
- Verifikation von parallel laufenden Teilaspekten ist nur selten möglich

Was können wir daraus lernen?

- Komplexe Systeme zu verifizieren ist schwierig bis unmöglich
- Wir können Fehler eindämmen aber nie ganz ausschließen

Was können wir daraus lernen?



- Softwareupdates können instabile Systeme erzeugen
- Es gibt keine *kleinen* Fehler

Was können wir daraus lernen?

- Wir sind heute von vielen komplexen Systemen abhängig
- Automatisierung sollte nicht zum Selbstzweck werden

Software-Fehler im Bereich Telekommunikation

USA, Lokales Telefonnetz, Juni/Juli 1991:

- Telefonnetz stundenlang lahm gelegt in Washington, Los Angeles und Pittsburgh (innerhalb einer Woche)
- Ursache: 3 falsche Bits in 2 Millionen-Zeichen-Programm

Software-Fehler im Bereich Telekommunikation

Notrufsystem England, März 1992:

- Programm der British Telecom, das Notrufe zur Ambulanz- Zentrale durchstellen und dort verteilen soll versagte
- Den Hilfesuchenden wurde nur "Bitte warten" vorgespielt
- Einige Anrufer verstarben während dieser Wartezeit

Software-Fehler im Bereich Telekommunikation

AOL wählte bei der Einwahl ins Internet (in den USA) den Notruf 911

- 9 für Amtsleitung
- 1170 warten auf Freizeichen deaktivieren
- dann die Rufnummer
- Beispiel: *911* 70 12345689

Software-Fehler im Bereich Telekommunikation

Schnurloses Telefon Kanada, 1993:

- Telefon wählt selbst durch zufällige Frequenzen Notruf 911

Quellen

- http://www.soft.com/AppNotes/attcrash.html
  - "Can We Trust Our Software?", Newsweek, 29.January 1990
  - ACM SIGSOFT, Software Engineering Notes, Vol.15, No. 2, Page 11ff, April 1990
- http://catless.ncl.ac.uk/Risks/9.63.html
- http://catless.ncl.ac.uk/Risks/9.62.html
- Bruce Sterling: The Hacker Crackdown
- http://www.att.com/spotlight/nethistory/