Největší softwarové katastrofy

Výrazy jako „nezdar, smůla a nešťastná náhoda“ jsou jen velmi mírné pro vyjádření toho, co už mají **CHYBNÉ PROGRAMY** na svědomí: havárie raket, zhroucení finančních burz a téměř i třetí světovou válku…

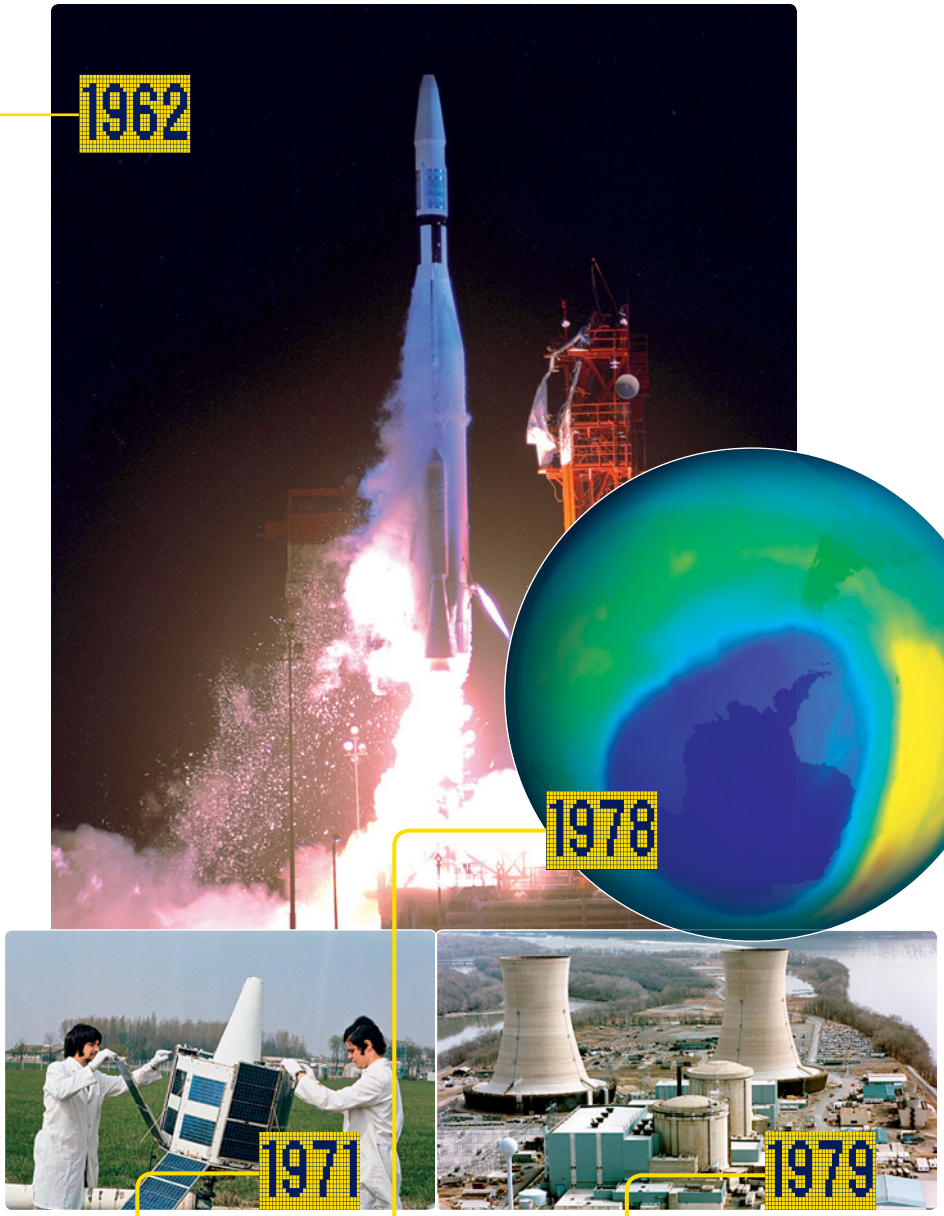
MARKUS MANDAU

22. července roku 1962 v 9 hodin 26 minut stiskl bezpečnostní specialista týmu na mysu Canaveral pověstný červený knoflík a zničil raketu Mariner. Tento čin představoval první z mnoha velkých budoucích softwarových katastrof.

Raketa Mariner 1 měla být první meziplanetární sondou, která prozkoumá Venuši. Místo toho se její trosky po letu trvajícím 293 sekund zřítily do Karibiku. O týden později objevili experti z NASA příčinu: programátor zapomněl opsat znaménko horního podtržítka (“¯“) z ručně psaného vzorce pro řídicí program. Tento symbol byl důležitý: měl zajistit, že počítač propočítá pozici a akceleraci z průměrné rychlosti. Protože tento symbol chyběl, počítač kalkuloval s nepřetržitě se aktualizujícími daty, která se měnila i během výpočtů. Aby se kolísání vyrovnalo, počítač neustále vysílal nové řídicí příkazy, kvůli kterým se však Mariner stále více odchyloval od zamýšlené trajektorie a ve finále musel být zničen. Poučení „z havárie programování“ se později objevilo v interní výzvě NASA: „Žádný detail není natolik malý, aby byl přehlédnut.“

Cesty do vesmíru: Havárie kvůli chybám

O 34 let později, 4. června roku 1996 v 9 hodin 34 minut, zaměstnanec řídicího pracoviště vesmírného centra v Kourou opět stiskl červené tlačítko a zničil Ariane 5 při



Mariner 1 Ihned po startu se začala raketa odchylovat od kurzu a musela být zničena. Důvod: Programátoři zapomněli ve vzorci na jediné znaménko.

Eole 1 Francouzský satelit způsobil zničení 71 meteorologických balonů. Důvod: Software nesprávně interpretoval žádost o předání naměřených údajů jako příkaz pro autodestrukci.

Nimbus 7 Satelit selhal při zkoumání ozonové díry nad Antarktidou. Důvod: Program pro analýzu získaných hodnot označoval neobvyklé naměřené hodnoty jako chyby.

Atomové reaktory Pět amerických jaderných reaktorů bylo vypnuto, když jim software pro kontrolu stability při zemětřesení dodal špatné údaje. Důvod: Program počítal místo součtu odmocnin součet mocnin.

jejím prvním letu. Po pouhých 37 sekundách se raketa odchýlila od svého kurzu a také ona musela být zničena. I její trosky nakonec skončily v Karibiku. Víceméně se očekávalo, že „linie problému“ bude podobná tomu u Marineru 1. Vyšetřovací komise Evropské vesmírné agentury ESA ale nakonec zjistila, že nehoda byla způsobena přetíženým navigačním systémem SRI, který inženýři převzali z Ariane 4 – zpráva tento krok zdůvodňovala takto: „Panoval obecný názor, že se nedoporučuje dělat jakékoliv změny v softwaru, který se v případě Ariane 4 výborně osvědčil.“ Tento přístup je obecně znám jako pravidlo „Nikdy nesahej do fungujícího systému“. Ariane 5 však nebyla jen větší, ale také přibližně pětkrát rychleji akcelerovala. Od SRI se očekávalo, že rychlost vypočítá snadno, protože se žádný problém neobjevil ani v případě Ariane 4. SRI ale nedokázalo konvertovat rychlost měřenou ve formě 64bitového čísla s pohyblivou čárkou do celočíselné 16bitové hodnoty – 16 bitů zkrátka k tomuto účelu nedostačuje.

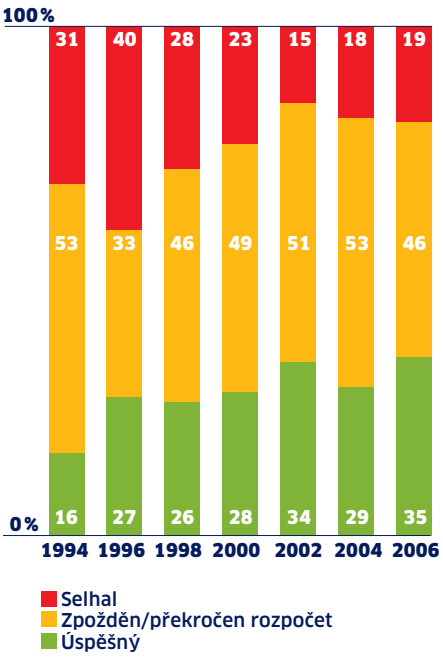
Výsledek: Přeplnění paměti, během kterého byly další hodnoty přepisovány. V případě vážného problému se předpokládalo, že se SRI vypne a budou se používat jen základní data z palubního počítače Ariane. Ten ale informace přijaté od SRI interpretoval jako navigační data (která vypadala, jako kdyby se raketa odchylovala od svého kurzu), a proto provedl fatální opravy kurzu, které nakonec vedly k havárii. Ve své zprávě došla ESA k závěru, že „rozhodující software by měl vždy podstoupit detailní testování“. Konkrétně: I jednoduchý test je dostatečný, aby tyto chyby objevil; SRI v Ariane 5 však „prošel“, aniž by byl i jen jedinkrát otestován.

Studená válka: Člověk versus stroj

Vesmírné katastrofy jsou sice okázalé, hororovými se však poruchy softwaru ukázaly během studené války. Obě supervelmoci totiž silně spoléhaly na počítačové analýzy, čímž vznikl velký prostor pro „katastrofické chyby“. Například v listopadu roku 1979 oznámil americký obranný systém NORAD celkem 2 020 současně útočících sovětských raket. Tento scénář byl tak nepravděpodobný, že armáda ihned usoudila, že se jedná o počítačovou chybu.

IT PROJEKTY: ČASTÉ SELHÁNÍ

V rámci svých CHAOS studií má „Standish Group“ od roku 1994 zmapováno více než 40 000 IT projektů. Výsledek: Míra úspěšnosti vzrostla pouze cca na jednu třetinu ze všech plánovaných projektů.



ZDROJ: WWW.STANDISHGROUP.COM

1982

Křižník HMS Sheffield Během války o Falklandy byla loď zasažena raketou a potopena. Důvod: Software přepnul zbraňové systémy do „safe“ modu.

1988

1983

Třetí světová válka Sovětský satelit detekoval pět mezikontinentálních raket. Podplukovník Petrov údaj vyhodnotil jako falešný poplach. Důvod: Software interpretoval světelné odrazy jako nepřátelské rakety.

1985–1987

Therac 25 Ozařovací přístroj zabil celou řadu pacientů nadměrnou dávkou radiace. Důvod: Software přístroje dokázal přesně řídit více procesů, pouze pokud byly příkazy zadávány pomalu.

1987

Wallstreet Zhroucení burzy vyústilo v denní ztrátu 22,6 procenta neboli 500 miliard amerických dolarů. Důvod: Burzovní software nedokázal zpracovat „stopsell objednávky“ dost rychle. To vyústilo v následné panické prodeje.

USS Vincennes Americký křižník sestřelil íránský Airbus – výsledkem bylo 290 mrtvých. Důvod: Podle šetření vyhodnotil systém Aegis (za přibližně 400 milionů dolarů) Airbus jako „předpokládaně nepřátelský“. Posádka věřila, že jde o útočící bojový letoun.

TOPSECRET

Plynovod Sibiřské plynovody nevydržely vysoký tlak a explodovaly. Důvod: SSSR získal řídicí program, který byl v USA „upraven“.

A příčina? V důsledku softwarové chyby propašoval počítač NORAD do vysílaných dat náhodné bity. Chyba poté infiltrovala zprávy, následkem čehož vzniklo „varování před útokem“.

Téměř vyhlazením lidstva skončil „špatně vyhodnocený“ signál z ruského satelitního systému. 26. září 1983 v půl jedné ráno signalizoval počítač vypuštění pěti amerických mezikontinentálních raket. Rozhodnutí, zda SSSR opravdu hrozí útok, bylo ponecháno podplukovníkovi Stanislavu Petrovovi, sedícímu v té chvíli asi 50 km jižně od Moskvy v utajeném bunkru „Serpuchov 15“. Jeho radu zvažoval generální štáb jako podklad k zahájení protiútoku.

Petrov se ale domníval, že jde o falešný poplach: „Měl jsem divný pocit v žaludku, a navíc – nikdo nezačíná nukleární válku s pouhými pěti raketami.“ Chyba byla detekována ještě ten samý den – satelit interpretoval odrazy slunce od mraků nad základnou vojenského letectva v Malstromu v Montaně (Malstrom Air Force Base) jako start raket.

Nedůvěřovat varováním počítačů jen proto, že se zdají nepravděpodobná, vás však také může přivést na scestí – jak se ukázalo při objevení ozonové díry. Pomocí měření, která prováděl na stanici v Halley Bay v Antarktidě, ji objevil britský badatel Joe Farman. V roce 1985 vydal svou zprávu v žurnálu „Nature“.

Později se zjistilo, že NASA ozonovou díru měřila již sedm let před ním pomocí satelitu Nimbus 7 – aniž by o tom kdokoliv věděl. Senzor satelitů TOMS zaznamenává fenomenálních 140 000 hodnot denně, avšak software analýzy, kterou NASA použila, byl naprogramován tak, že označil neobvyklá měření pod 180 Dobsonových jednotek jako „neplatná“, přičemž mezery dat projevující se jako výsledek měly být vyplněny „více pravděpodobnými“ hodnotami. Původní hodnoty pak bez povšimnutí skončily v databázi až do doby, kdy byl publikován Farmanův článek. Krátce po jeho zveřejnění mohla i NASA poreferovat o ozonové díře; stačilo jen ve zprávě použít své dřívější „chyby v měření“. Se správně nastaveným softwarem mohl být objev připsán NASA…

Velké projekty: Platforma pro nehody

Ve srovnání s „průšvihy“ 20. století byly softwarové „katastrofy“ nedávných let spíše neškodné. Kvůli nehodám sice selhaly velké projekty (které plnily titulky novin), ale lidský život v nich nebyl ohrožen. V případě Airbusu A380 byla „v nebezpečí“ maximálně EADS, jedna z největších evropských společností působících v leteckém a kosmickém průmyslu, a to kvůli mrzkému důvodu – příliš krátkým kabelům.

V současnosti největší osobní dopravní letadlo světa bylo kompletně navrženo pomocí softwaru a zároveň jde o jeden z největších mezinárodních projektů. A právě zde začínají problémy: elektronika včetně kabeláže byla vyrobena v Hamburku, zatímco trup letadla byl vyroben v Toulouse. V létě 2004, když technici začali montovat kabely do letadel (jejich celková délka byla 530 km), vyšlo najevo, že mnoho kabelů je příliš krátkých. Náklady se zvýšily, dodací termíny byly „upraveny“ – a problémy narostly až do takové výše, že je management Airbusu musel v roce 2005 veřejně přiznat…

Jedna z analýz uvedených Christianem Streiffem, toho času šéfem Airbusu, objevila problém: mechanici použili rozdílné verze softwaru CATIA, které nebyly kompatibilní (jmenovitě v Hamburku použili verzi 4, zatímco v Toulouse verzi 5).

Ačkoliv se tato neshoda verzí může zdát jako neškodná, jsou mezi těmito verzemi obrovské rozdíly. Verze 4 je psána ve Fortranu a běží na pracovištích založených na Unixu, verze 5 pak byla nově vyvinuta v C++ a v Toulouse běžela na počítačích s Windows. Rozdíly měly za násle-

1990 at&t

AT&T Řetězová reakce nového telefonního softwaru přivedla všechna kontrolní centra do „fáze resetu“; v důsledku toho nebyly spoje v provozu více než devět hodin. Důvod: Byl chybně spuštěn příkaz „break“.

1991

Patriot Obranný systém nedetekoval rakety Scud, což mělo za následek smrt 28 vojáků. Důvod: Výpočetní chyba v časovém měření. Čím pracoval systém déle, tím větší byla chyba.

1995

Letiště Denver Nový automatický systém pro manipulaci se zavazadly se zcela zhroutil. Důvod: Příliš složitý systém specifikací přetížil software.

1996

Ariane 5 Raketa se odchýlila od svého kurzu a musela být zničena. Důvod: Přetečení bufferu v softwaru použitém z předchozí rakety Ariane 4.

1999

Umělá družice pro Mars Sonda shořela v atmosféře Marsu. Důvod: Součástí sondy byly britské přístroje měřící tlak v librách, zatímco NASA používá metrický systém.

dek také nekompatibilní formát souboru. Není možné konvertovat soubory beze ztrát; metadata, jako jsou vysvětlivky od vývojáře ke komponentě, byly obvykle při konverzi ztraceny. Ačkoli Airbus obstaral řádný konverzní software pro sdílenou databázi, nefungoval tak, jak se očekávalo, a proto byl zřídka používán.

Podle Streiffa se skutečný problém nacházel v „horních" úrovních: „Bylo to kvůli selhání managementu při zajištění účinné spolupráce v projektu." John Leahy, senior obchodní ředitel Airbusu, podotkl: „Úřady jednoduše dělaly, že to nevidí." Selhaly při zaškolení dostatečných techniků pro CATIA 5 a selhaly také při zahánění nelibosti vůči novému softwaru v Hamburku. Výsledkem pak byly další náklady v hodnotě pěti miliard eur.

Pozor: Když software myslí

Německé správní orgány nemají dobré zkušenosti s IT projekty, a to od vybírání mýta až po elektronické zdravotní karty. Například v případě projektu Hartz IV (projekt na podporu zaměstnanosti) měl software řadu závad; program A2LL (webová aplikace pro správu finančních služeb) nikdy nefungoval bez chyb. Vyznamenal se především v zimě roku 2004, kdy statisíce příjemců financí z programu Hartz nezískaly žádné peníze. Příčina byla triviální: krátká čísla kont vyplnil A2LL na konci nulami – aby měl deset „požadovaných" číslic. Software byl ale naprogramován špatně – nuly měly být správně doplněny na začátek čísla konta. Například poštovní banka poté musela zřídit zvláštní krizové manažerské skupiny, aby vrátila peníze manuálně.

Od té doby je A2LL, společně s dalším softwarem vyvinutým firmou T-Systems, v příslušných úřadech považován za ztracený případ. To však není konec celé frašky – úřad nyní plánuje nový software zvaný „Allegro". Projekt se odhaduje na pět let a počítá s náklady okolo 90 milionů eur.

AUTOR@CHIP.CZ

NEJČASTĚJŠÍ SOFTWAROVÉ CHYBY

Na softwarové chyby lze narazit i v kvalitně otestovaných a často používaných programech. Během několika let testoval softwarový „provider" Coverity více než 150 velkých open-source projektů pomocí vlastního nástroje založeného na bázi statické analýzy kódu. Zmiňované projekty byly určeny především pro Linux a server Apache. Celkem software analyzoval více než 55 milionů řádků kódu.

Před třemi roky odhalil jednu chybu na 3 300 řádků, v současnosti je tato hranice posunuta až na 4 000 řádků. Mezi nejčastěji detekované chyby patřily tyto:

Typ chyby	Výskyt
Ukazatel nastavený na null odkazuje do oblasti paměti	27,95 %
Použití neschválené oblasti paměti	25,73 %
Nepřístupný programový kód	9,76 %
Použití netestovaných proměnných	8,30 %
Přetečení staticky adresované oblasti v paměti	6,14 %
Použití již uvolněné paměti	6,46 %
Nepovolené počítání s nulou (např. dělení)	5,85 %
Proměnným nejsou přiřazeny hodnoty	5,50 %
Vypočítané záporné hodnoty jsou použity bez kontroly	3,72 %
Ukazatel odkazuje na paměť, která není přístupná	0,62 %
Přetečení dynamicky adresované oblasti v paměti	0,31 %

ZDROJ: SCAN.COVERTY.COM

Sonda na Mars Sonda určená k přistání na Marsu byla zničena přistáním v rychlosti 80 km/h. Důvod: Software detekoval přistávací „nohy" sondy jako povrch Marsu a vypnul motory.

Výpadek Více než 50 milionů domů v USA a Kanadě postihl výpadek elektrického proudu. Důvod: V softwaru na řízení rozvodů elektřiny selhal výstražný systém.

Hartz IV Statisíce příjemců financí z německého „sociálního" systému neobdržely ani euro. Důvod: Program doplnil u čísel účtů nuly na špatné straně.

Airbus A380 Gigantické letadlo, jehož vývoj stál více než pět miliard eur, bylo značně „zpožděno". Důvod: Vývojáři používali nekompatibilní verze CAD systému CATIA.

BNP Paribas Software banky „vyplenil" desítky tisíc účtů pomocí 600 000 opakovaných transakcí. Důvod: Doposud není znám...