ČLOVEK AKO PÔVODCA A RIEŠITEĽ PROBLÉMOV

Pokiaľ ani jednoduché systémy nedokážu fungovať bez občasnej chyby, tak ako môžeme očakávať dokonalosť od niečoho tak zložitého, ako je ľudská myseľ.

Ivan Valenčík

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
valencik[zavináč]gmail[.]com

Abstrakt. *Vetu vraviacu „Mýliť sa je ľudské“ počul pravdepodobne každý z nás. Nečudo, veď každý zažil určite množstvo vlastných alebo cudzích pochybení. Čím zložitejšej činnosti sa človek venuje, tým náchylnejší je k tomu, že spraví niečo neželané. Softvérové systémy neustále naberajú na svojej komplexnosti - nové oblasti aplikácie, distribuovanosť, rozšíriteľnosť a nárast veľkosti nasadenia sú len niektoré z pribúdajúcich výziev pre vývojárov. So zväčšujúcou sa zložitosťou vznikajú aj ďalšie riziká a ich zmiernenie má zase na starosti človek. Pri tom väčšinu naplnených rizík má na svedomí „ľudský faktor“. Dôvodov k týmto zlyhaniam je mnoho, od slabšieho vnímania až k nedostatku skúseností alebo vedomostí. Možných riešení, líšiacich sa od seba svojou náročnosťou a efektivitou, je tiež veľké množstvo. Preto je nad tým, ako sa vyhýbať človekom spôsobeným rizikám, veľmi dôležité uvažovať. Už od Senecu totiž vieme, že mýliť sa je ľudské, ale zotrvávať v omyle je diabolské.*

Kľúčové slová: *ľudský faktor, riziko, manažment rizík*

Kde vlastne vzniká pri tvorbe softvéru riziko?

Oblasť výpočtovej techniky sa rozvíja a mení svet rýchlosťou, ktorá je v ľudskej histórii nevídaná. V priebehu niekoľkých dekád sa úplne zmenil spôsob, ako medzi sebou

komunikujeme, aké informácie máme k dispozícii a hlavne, ako rýchlo sa k nám dostanú, počítače nám umožňujú vytvárať zariadenia, ktoré by naši predkovia považovali za mágiu – počnúc mobilným telefónom, idúc cez prístroje dovoľujúce nahliadnuť do útrob ľudského tela bez jeho poškodenia a končiac trvalou prítomnosťou človeka na obežnej dráhe Zeme. Pokiaľ sa dokážeme na nejaký čas zamyslieť nad tým, aké komplexné riešenia sa skrývajú za vecami, ktoré sú pre nás bežné, tak nás musí nutne uchytiť úžas. Množstvo ľudského dôvtipu potrebného na vytvorenie bežného dnešného stolného počítača, s jeho viacjadrovým procesorom a aplikáciami riešiacimi nespočet úloh je enormné.

Tvorba softvéru je vo svojej podstate vlastne zápas so zložitosťou - vývojár sa pri nej snaží prostredníctvom rôznych technológií zachytiť do svojho diela rozličné myšlienky, matematické poznatky, vlastnosti systémov a vlastne hocičo, čo sa od výsledku jeho práce očakáva. Jeho úloha sa pri tom s novými postupmi a nárokmi stále sťažuje. Nové aplikácie majú riešiť väčšie problémy, na ich zvládnutie musia byť distribuované, mali by byť rozšíriteľné, efektívne a hotové najlepšie ihneď. Na ich vytvorenie je potrebné odborníkov z viacerých oblastí, čo do už tak zložitej situácie pridáva ešte aj problém vzájomnej komunikácie a organizácie.

Zložitosť týchto výziev je často tak veľká, že pri analyzovaní rizík, ktoré zo sebou projekt prináša, sa často sústredíme iba na technické aspekty problému. Tie sú síce bez pochyby dôležité, avšak nemali by sme zabúdať na to, že prirodzenou ľudskou vlastnosťou je mýliť sa a aj napriek všetkej snahe sa táto vlastnosť nutne premietne do vykonávanej práce. Veď pokiaľ vieme, že deväť z desiatich autonehôd spôsobí ľudský faktor a rovnaký pomer platí aj pre vzniknuté stavy ohrozenia v jadrových zariadeniach [5], tak nemáme dôvod si myslieť, že tvorba softvéru by mohla byť výnimkou.

Môžeme zodpovednosť priradiť čipu?

Pri hlbšom zamyslení sa nad rizikami vyplývajúcimi z technológie, či už je to hardvér alebo softvér, taktiež často môžeme prísť k záveru, že skutočné riziko sa nachádza medzi klávesnicou a stoličkou. Iste, mnohým situáciám sa nedá predísť, veď napríklad vnútornú chybu materiálu súčiastky si objektívne nie je možné všimnúť, ale veľa technických problémov vznikne len vďaka nevhodnému spôsobu nasadenia alebo manipulácie, či kvôli nedostatočnej kontrole. Chybný kód, ktorý je príčinou naplneného rizika, totiž niekto musel napísať, a potom sa nechať dostať do situácie, v ktorej mal možnosť zlyhať. Problematika určite teda nie je taká jednoduchá, že by sme mohli riziká v softvérových projektoch priradiť len k ľudom, ale zdá sa, že tejto kategórii ohrození je mnohokrát venovaná príliš malá pozornosť.

Problém ľudskej nespoľahlivosti sa javí byť o to závažnejší, že za zmierňovanie systémových, technických a aj ľudských rizík je často zodpovedný znova človek. Pôvodca potenciálneho pochybenia má teda zároveň dozerať na to, že k nemu nedôjde. To, že sa možno nejedná o tú istú osobu, pri tom nemusí byť vôbec ukľudňujúce.

A oplatí sa týmto vôbec zaoberať?

Analýza rizík je pri mnohých softvérových projektoch opomenutá, keďže nie je nutnou podmienkou na vypracovanie projektu. Nad tým, či ju je potrebné vypracovať, sa je

potrebné vážne zamyslieť, keďže sa môže stať, že na spracovanie rizika bude vynaložená príliš veľká miera úsilia v pomere k tomu, aké úsilie by bolo potrebné vynaložiť v prípade naplnenia rizika [4]. Odhadnúť mieru rizika spôsobenú človekom môže byť totiž veľmi zložité, rovnako ako navrhnúť vhodné postupy na jeho zmiernenie.

Prvá otázka ktorá sa ponúka je, na čo sa vlastne zamerať? Analyzovaný systém je ovplyvňovaný vnútornými, ako aj vonkajšími vplyvmi. Ďalej sa v systéme môže nachádzať zložitá sieť prepojení navzájom sa ovplyvňujúcich ľudí, procedúr a vybavenia na rôznych úrovniach. A v tomto komplexnom prostredí sa musíme rozhodnúť, ktoré riziká rozpracujeme, pričom máme na výber z jasných rizík, ako sú napríklad pochybenia konkrétnych osôb pri konkrétnych úlohách, ale aj ťažšie uchopiteľné riziká, ku ktorým sa dá priradiť napríklad kultúra organizácie [5].

Ďalšia vyvstávajúca otázka je, či vôbec sme schopní riešiť odhalené riziko. Zatiaľ čo s antropometrickými otázkami, teda s otázkami zaoberajúcimi sa fyzickými aspektmi systému - vzťah k ľudskej veľkosti, zraku, sluchu atď., si vieme celkom dobre poradiť, venovať sa kognitívnym ťažkostiam je omnoho náročnejšie a menej objektívne, keďže myseľ zostáva pre vývojára skrytá [5].

Pri vývoji softvéru sme často ochotní tolerovať relatívne veľké množstvo chýb a nedostatkov. Tie nebývavajú všetky odstránené dokonca ani vo finálnom produkte. Pri tom dôvod nie je ani tak to, že by sme ich neboli schopní postupne poodstraňovať, ale skôr to, že ich vychytanie nie je hodné prostriedkov, ktoré by museli byť naň vynaložené. Určité množstvo pochybení si môžeme väčšinou dovoliť taktiež v procese vývoja.

Takýto postoj k projektu však nie je možné zaujať vždy. Niektoré systémy musia fungovať bezchybne, ich zlyhanie by malo príliš vážne dôsledky. Príkladom takéhoto diela môže byť systém na riadenie letovej prevádzky alebo softvér v lekárskych prístrojoch, teda hlavne technológie, na ktorých sú priamo závislé ľudské životy. Rovnako kritický môže byť aj spôsob tvorby a údržby systémov, napríklad výpočty potrebné na riadenie chemickej výroby alebo jadrovej reakcie nie je predsa možné skúšať priamo na prebiehajúcom procese. Kritický ja často tiež čas, dokedy musí byť projekt dokončený. Spomenuté príklady sú akýmsi extrémom, kde zlyhanie nie je prípustné vôbec, ale podobné požiadavky často vznesú aj spoločnosti, ktoré by síce životy ľudí priamo neohrozili, ale vzniknuté ťažkosti sú pre nich príliš nákladné, aby si ich mohli dovoliť. V týchto prípadoch je analýza rizík nutná.

A keďže od takejto analýzy očakávame, že odhalí všetky významné riziká, tak sa musí človekom zaoberať podrobne. S týmto názorom je ťažko v kontexte štatistiky hovoriacej o tom, koľko nehôd je zavinených ľudským pričinením, nesúhlasiť. Nutné je však uvedomiť si celú šírku problému a neopomenúť vplyv žiadneho účastníka v procese. Vplyv na úspech softvérového projektu nemajú zďaleka len vývojári, ale vo veľmi veľkej miere ľudia, ktorí ho manažujú, používajú, či chcú na neho z nejakého dôvodu útočiť.

Čo všetko môžeme pokaziť?

Skúsme sa teda pozrieť na riziká spojené s ľudským faktorom bližšie. Podľa [3] roly, ktoré jednotlivec pri vývoji softvéru zastáva, spadajú do niektorej z nasledujúcich kategórií: individuálne, tímové, riadiace a zainteresované. Zaradenie roly nemusí byť vôbec

jednoznačné, je prirodzené, že dochádza k prekryvom. Niektoré rizikové faktory individuálnych a tímových rolí sú nasledovné:

- osobná schopnosť osvojiť si vývojové metódy, jazyk a nástroje
- skúsenosti a vodcovstvo vedúceho tímu
- výkon tímu
- dostupnosť skúsených pracovníkov
- oddanosť projektu
- osobná lojalita pracovníkov k spoločnosti
- skúsenosť s identifikovaním a analyzovaním rizík v manažmente rizík atď.

Manažéri a ďalší zainteresovaní aktéri majú na riziká projektu odlišný náhľad. Predmetom ich záujmu je dosiahnutý zisk, dodržanie časového harmonogramu, kvalita, efektivita pracovníkov a pod. Objednávateľovi záleží na návratnosti investície, úrovni bezpečnosti, efektivite aplikácie a jej použiteľnosti. Pri analýze rizík súvisiacich s riadením nesmieme zabúdať na rozdielne motivácie zúčastnených strán a na ich odlišnú znalosť problematiky. Správne manažérske rozhodnutia majú na úspech projektu zásadný vplyv, takže ich podcenenie môže v konečnom výsledku vyústiť v katastrofu. K najdôležitejším faktorom, ktoré treba zobrať do úvahy patrí:

- rozhodnutia a podpora manažmentu
- úroveň sebavedomia manažmentu
- výber správnych pracovníkov
- spolupráca s externými organizáciami
- dohody medzi manažmentom a poskytovateľmi služieb
- vhodné školiace zdroje
- pravidelný dohľad nad bezpečnostnými pravidlami a procedúrami
- podpora efektívneho aplikovania pokročilých metodológií
- pravidelné zhodnotenie rizík a plánovanie bezpečnosti
- udržiavanie rezervných zdrojov a času na vysporiadanie sa s rizikom atď.

Za nemenej dôležitú je treba považovať aj organizáciu pozícií a ich vzájomné vzťahy, ktorá priamo ovplyvňujú procesy, náklady a harmonogram vývoja softvéru. Niektoré riziká spojené s organizáciou sú tieto:

- organizačná štruktúra a jej stabilita
- interná a externá komunikácia
- efektívnosť
- vyspelosť
- prostredie na zavedenie bezpečnostných pravidiel a procedúr
- integrácia bezpečnostných otázok s každodennými činnosťami
- vhodné priestory pre vývoj softvéru
- súlad s právnymi požiadavkami atď.

Vidíme teda, že vplyv ľudského faktoru je veľmi široký. Okrem toho je ťažké nájsť jednoznačne správne postupy, keďže riziká navzájom súvisia. Ako príklad môžeme uviesť vysporiadanie sa s rizikom, že programátor nedodá potrebnú funkcionalitu v požadovanej

kvalite. Riešením tohto problému môže byť jej priebežné sledovanie a v prípade, že vývoj speje k neúspechu, pridelenie ďalšieho špecialistu k tomuto problému. To bude mať pravdepodobne za následok, že sa spomalí postup na úlohách, ktoré riešil doteraz a nutné zvýšenie pravdepodobnosti nenaplnenia jeho ďalších povinností. Ďalej by sme si mali uvedomiť, že náš problém znova rieši len človek a stále si nemôžeme byť úplne istí, že dokáže zvrátiť nepriaznivý vývoj. To však nie je prekvapivé, koniec koncov nehovoríme o odstraňovaní rizík, ale len o ich zmierňovaní.

Aj v tomto príklade sme si mohli všimnúť, že veľmi záleží na vedomostiach, schopnostiach a charakterových vlastnostiach ľudí zainteresovaných vo vývoji softvérového produktu. Nevhodné pridelenie úloh je rýchlou cestou k neúspechu. K nemu často vedú nesprávne očakávania. Príkladom môže byť vývoj aplikácie, ktorá musí spĺňať určité bezpečnostné požiadavky. V návrhu a pri vytváraní riešenia sa často predpokladá, že koncový používateľ nie je technicky zdatný a aplikácia je vytváraná tak, aby mu neumožnila robiť chyby. To je úplne v poriadku a výsledkom by mala byť vysoká bezpečnosť ponúknutého riešenia. Na druhej strane od vývojára sa očakáva vysoká odbornosť a predpokladá sa, že dodrží všetky postupy vedúce k zabezpečenému riešeniu. K dispozícii pri tom zvykne mať všetky prostriedky bez obmedzenia. Pri tom je naivné predpokladať, že napríklad odborník na užívateľské prostredia bude zároveň aj odborníkom na ostatné aspekty bezpečnosti aplikácie. To, že nie každý je zároveň aj expertom na bezpečnosť je úplne prirodzené, veď od bezpečnostných expertov sa tiež nevyžaduje, aby boli zároveň odborníkmi na grafiku. Práca vývojárov s minimálnymi vedomosťami o bezpečnostných rizikách napriek tomu často nie je dostatočne skontrolovaná [6].

Osobitou kategóriou rizika, ktorá sa stáva v globalizovanom svete výraznejšou sú kultúrne rozdiely. Manažment rizík je v úzkom prepojení s kultúrnym kontextom vývojových aktivít. Každý človek sa nachádza vo svojom sociálnom a kultúrnom prostredí, ktoré má nepopierateľný vplyv na to, ako človek rozmýšľa a koná. Tieto rozdiely si nemusíme uvedomiť pri práci v rámci regiónu, ale stávajú sa významnými, pokiaľ spoločnosť pôsobí nadnárodne alebo dokonca globálne.

V odlišných kultúrach môžu mať zamestnanci podstatne odlišný vzťah k nadriadeným, schopnosť reagovať na nejasné situácie, teda odlišný prístup k preberaniu zodpovednosti, iné očakávania v prejavovaní vlastnej iniciatívy, môže prevládať viacej kolektivizmus alebo individualizmus a môže sa jednať až o tak výrazné rozdiely, akým je napríklad postavenie žien v spoločnosti. Individuálne ľudské povahy sú veľmi rozmanité, ale ľudia prichádzajúci z toho istého kultúrneho prostredia zvyknú mať spoločné charakteristiky, ktoré sa po svojom sformovaní už zvyknú veľmi ťažko meniť. Pri projektoch, kde sa stretávajú pracovníci z rôzneho kultúrneho prostredia, je teda vhodné zaoberať sa aj charakteristikami týchto prostredí, hlavne ich odlišnosťami, čo by po správne vykonanej analýze rizík malo mať za následok vyhnutie sa možným konfliktným situáciám [1].

Ako sa teda s rizikom vysporiadať?

Identifikácia a ohodnotenie rizík je síce neoddeliteľnou súčasťou manažmentu rizík, ale jeho hlavným účelom je ich prioritizovanie a určovanie postupov na ich zmiernenie.

Keďže riziká vyplývajúce z ľudského faktoru sú veľmi rôznorodé, tak aj spôsoby na ich odstránenie sa navzájom výrazne odlišujú. Okrem toho existuje ku každému riziku viacero prístupov na to, ako sa s ním vysporiadať. Efektivita riešení jedného problému zvykne byť od seba veľmi výrazne odlišná a výber nesprávneho spôsobu na pokrytie rizika sa môže stať pre projekt osudným. Riešenia hrozieb sa od seba líšia ich nárokmi na zdroje, čiže majú iné nároky na čas, pracovníkov s rôznymi vedomosťami a materiálne prostriedky. Ďalším dôležitým faktorom, ktorý je rozhodujúcim pri výbere vhodného postupu, sú vedľajšie efekty, ktoré z prípadnej nutnosti venovať sa riziku vyplynú. A nakoniec samotný efekt zvoleného riešenia môže mať mnoho podôb. Je veľký rozdiel, či dôsledky rizika odstránime úplne, len ich zmiernime alebo celú situáciu zmeníme od základu, čiže sa nás už jeho efekty nebudú týkať.

Jeden spôsob, ako sa vysporiadať s rizikom, je sa mu vyhnúť. Snažíme sa ho pri tom úplne eliminovať tým, že sa z rizikovej činnosti stiahneme alebo sa jej vôbec nezúčastníme. Tento spôsob sa javí byť vo väčšine prípadov relatívne jednoduchý na vykonanie, avšak pri ňom ľahko prídeme o príležitosti. Je pravda, že kto nehrá, tak nič neprehrá, ale nesmieme zabudnúť, že ani nič nevyhrá a na rozdiel od hazardu, pri softvérových projektoch by sme mali mať svoje šťastie vo svojich rukách. Preto si je potrebné zodpovedne premyslieť, či upustiť od svojich pôvodných plánov alebo nájsť spôsob ako v nich pokračovať.

Pokiaľ sa riziko rozhodneme podstúpiť, tak by sme sa mali pokúsiť nájsť spôsob, ako ho zredukovať. To sa dá dosiahnuť optimalizáciou našich činností, či zmierňovaním jeho dôsledkov. Určite by sme boli najradšej, keby sme vždy dokázali zabezpečiť, že riziko úplne odstránime, ale to nie je možné, takže sa musíme naučiť byť spokojní aj s neúplnými riešeniami. Význam zmierňovania rizík by bolo veľmi nešťastné podceniť, keďže niekoľkými opatreniami dokážeme často docieliť, že dôsledky naplneného rizika budú podstatne menšie, ako keby sme sa mu vôbec nevenovali a zastihlo by nás nepripravených. Múdrosť o tom, že pripraveným šťastie praje, nie je nadarmo predávaná z generácie na generáciu.

Pokiaľ sú pre nás nástrahy projektu príliš veľké, tak im nemusíme čeliť sami, ale môžeme sa o ne podeliť. Zdieľať riziko síce nebude nikto len tak, ale existuje mnoho spôsobov jeho prenesenia, ktoré sú výhodné pre obe strany. Pri riešení úloh, ktoré sú mimo nášho zamerania alebo ich z iného dôvodu práve nedokážeme zvládnuť, sa dá práca outsourcovať. To prináša samozrejme ďalšie riziká, ale pri dobre spravenej dohode má outsourcing potenciál zbaviť nás nástrah, ktoré by boli inak veľmi nákladné alebo priam neuskutočniteľné.

Ďalší spôsob na to, ako sa s niekým podeliť o riziko, je poistenie. Jeho nutným následkom sú vyššie náklady, ale pri niektorých činnostiach je veľmi dôležité. Vhodné je ho využiť hlavne v situáciách, keď je riziko vzniknutia poistnej situácie veľmi malé, ale následky z nej by boli likvidačné. Okrem jeho hlavného účelu, teda krytia škôd v prípade, že príde k naplneniu rizika, stojí aj za zmienku aj to, že pokiaľ sme poistení, tak sa v rizikových situáciách nemusíme obávať každého svojho kroku, čo sa prejaví vo vyššej produktivite.

K riziku sa dá postaviť aj tak, že ho budeme jednoducho akceptovať. Činnosť s ním spojená totiž pre nás môže byť stále atraktívna a vyhýbať sa mu alebo ovplyvňovať jeho dôsledky nemusí byť možné alebo výhodné. Aj tak je užitočné o ňom vedieť, úplne

nečakané údery bývajú totiž najnepríjemnejšie. Získame príležitosť toto riziko priebežne sledovať a prípadne k nemu prehodnotiť postoj. Akceptovať riziko neznamená ignorovať ho, reakcia môže byť napríklad vytvorenie si finančnej alebo časovej rezervy.

Zvládanie rizík je zodpovednosťou manažmentu, ale vyžaduje si účasť všetkých zamestnancov, a to vo všetkých fázach. Vedúci si totiž nemusí a často vlastne ani nemôže uvedomiť všetky projektu hroziace situácie. Pritom by bezpečnostné pravidlá a postupy navrhnuté vedením mali pokrývať pôsobenie celej spoločnosti. Je dôležité, aby pri návrhu týchto opatrení boli zobrané do úvahy všetky osobné roly a povinnosti, kritéria pre akceptovateľnú mieru rizika a vytvorili nástroje na zavedenie, monitorovanie a prehodnocovanie úrovní rozpoznaných rizík. Po rozoznaní rizík majúcich za príčinu ľudský faktor je vhodné zamyslieť sa aj nad týmito prostriedkami [3]:

- bezpečnostné pravidlá
- pravidlá prístupu
- zodpovednosť používateľov
- metodológia manažmentu rizík
- organizačná štruktúra
- identifikácia, zaradenie a povolené využitie zdrojov
- overovanie vstupov a výstupov
- kryptografická kontrola
- fyzické zabezpečenie prostredia
- detekovanie záškodníckej aktivity a monitorovanie siete
- detekovanie škodlivého kódu pri vývoji
- bezpečná architektúra, návrh, kódovanie a testovanie
- určenie potrebných školení, ich vykonanie a vyhodnotenie
- zhrnutie schopností a skúseností všetkých zamestnancov
- priebežné vzdelávanie o bezpečnosti
- zoznam činností na vykonanie pri rozširovaní a údržbe systému
- pripravené reakcie na narušenia bezpečnosti
- súlad s právnymi normami, atď.

Riešenie treba vybrať s ohľadom na konkrétnu situáciu a firemné prostredie. Pri zvolení správnych postupov a ich doplnení o vhodné technologické vybavenie je predpoklad, že úspešne splníme vytýčené úlohy. Nikdy by sme nemali pri tom zabúdať na to, že pre dosiahnutie výnimočného výsledku musíme dbať na každý detail. Pre plné uvedomenie si tejto skutočnosti si pripomeňme udalosť, o ktorej sa často hovorí ako najdrahšom softvérovom bugu. 4. júna 1996 sa mal uskutočniť prvý let rakety typu Ariane 5. Na jej ovládanie bol prevzatý kód z rakety Ariane 4, avšak ich odlišnosti neboli dostatočne zobrané do úvahy. Motory Ariane 5 majú pri tom väčší výkon, čo malo za následok, že pri jednej z konverzií čísiel došlo k pretečeniu, ktoré zase spôsobilo zlyhanie celého počítača na ovládanie letu. Tým pádom chyba, s ktorou má skúsenosť každý, kto sa aspoň nejaký čas venoval programovaniu, spôsobila, že zhruba po 40-tich sekundách sa raketa hodná stoviek miliónov eur sama zničila [2]. Táto udalosť by nám mala pripomínať, že v kritických projektoch nesmieme podceniť žiadne riziko, lebo reťazenie nepriaznivých udalostí môže vyústiť do skutočnej katastrofy.

Čo sme si teda povedali?

Softvérové systémy sú človekom vytvorené diela, teda ich kvalita je priamo závislá od kvality ľudskej práce, ktorá bola do nich vložená. Vo výsledku sa teda okrem talentu vývojárov prejavia aj ich nedostatky a zlé rozhodnutia. Manažment rizík umožňuje prejavom týchto neželaných skutočností čeliť. Pri práci s technológiami sa síce zväčša sústredíme na ne, takže ľudský faktor je často podcenený, či už preto, že sa od všetkých očakáva zvládnutie svojich úloh alebo len z nevedomosti. Táto skutočnosť nie je ani veľmi prekvapivá, veď ľudská myseľ je aj napriek všetkým zázrakom modernej vedy stále záhadou.

Narastajúcou zložitosťou vytváraných systémom a s pribúdajúcimi oblasťami ich nasadenia sa objavujú stále nové výzvy, ale vývoj ide dopredu, keďže zároveň sa učíme zvládať tie staré čoraz lepšie. A zatiaľ čo technické riziká sa postupne menia, tie ľudské zostávajú a pokiaľ nedôjde k úplne nečakanému pokroku v umelej inteligencii, tak nikdy nezmiznú. Z množstva spôsobov, ako na nejaké riziko reagovať, je pre človeka veľmi náročné vybrať a stále sa nič nemení na tom, že ani on nemusí spraviť správne rozhodnutie. Práve pri uvažovaní nad ľudským správaním, aj keď v kontexte s najnovšími výdobytkami ľudského poznania, vidíme, že človek sa od dôb starovekého Ríma v niektorých veciach vôbec nezmenil. Stačí si pripomenúť myšlienku Senecu, ktorý povedal: „Errare humanum est, sed perseverare diabolicum“, čiže: „Mýliť sa je ľudské, ale zotrvávať v omyle diabolské“.

Použitá literatúra

1. Ma, W., Liu, L., Feng, W., Shan, Y., Peng, F.: *Analyzing project risks within a cultural and organizational setting*. IEEE Computer Society, Washington, D.C., 2009. Dostupné online: http://dx.doi.org/10.1109/LMSA.2009.5074858 (2010-10-04)
2. Lions, J.L. et al.: *ARIANE 5: Flight 501 Failure*. Report by the Inquiry Board, Paris, 1996. Dostupné online: http://sunnyday.mit.edu/accidents/Ariane5accidentreport.html (2010-10-18)
3. Islam, S., Dong, W.: *Human factors in Software Security Risk Management*. ACM, New York, 2008. Dostupné online: http://doi.acm.org/10.1145/1373307.1373312 (2010-10-18)
4. Williams, R.C.; Walker, J.A.; Dorofee, A.J.; *Putting Risk Management into Practice*. IEEE Software, vol. 14, no. 3, pp. 75-82, May/June 1997. Dostupné online: http://doi.ieeecomputersociety.org/10.1109/52.589240
5. Sandom, C.: *Success and Failure: Human as Hero – Human as Hazard*. Australian Computer Society, Inc., Darlinghurst, 2007. Dostupné online: http://portal.acm.org/citation.cfm?id=1387040.1387049 (2010-10-18)
6. Wurster, G., Oorschot P.C.: *The Developer is the Enemy*. ACM, New York, 2009. Dostupné online: http://doi.acm.org/10.1145/1595676.1595691 (2010-10-18)

Annotation

Human as an Originator and as a Solver of Problems

Everyone has surely heard a quote "to err is human". No wonder, because everyone must have experienced a number of mistakes of their own or other's. With an increasing complexity of the task at hand increases also the probability that something will go wrong. Software systems are continuously becoming more complicated – some of the new challenges are the need for distribution, extensibility, scale of the system and new areas of application. With an increasing complexity are coming new risks and it is once again human responsibility to mitigate them. On the other hand "human factor" is responsible for most of the incidents. Reasons for this are diverse, from the lower perception to the lack of knowledge or skill. There are also many kinds of solutions, some of them more suitable than other. That is why it is so important to contemplate on how to avoid human caused risks. It is a long time since Seneca said to err is human, but to persist in the mistake is diabolical, but it is still very true.