Lze rozeznat člověka od počítače?

Zpátky k Turingovi...

Pozn.: Článek byl otištěn v časopise Data Security Management (č. 2, sv. 10) a je zpřístupněn s jeho souhlasem.

Stále častěji se můžeme na Internetu setkat s případy využívání (či zneužívání) veřejně přístupných služeb, původně zamýšlených pouze pro interakci s člověkem, pomocí robotů. Jak se dnes tento problém automatizovaného zneužívání jednoduchosti přístupu ke službám řeší?

Nejznámější instancí tohoto problému je spam – nevyžádaná pošta, která si našla cestu do schránky snad každého z nás. Kromě e-mailového spamu se však objevují i jiné formy elektronického „obtěžování“, které využívají možnost neautentizovaného zasílání příspěvků do veřejných fór, přidávání komentářů k článkům na velkém množství blogů ap. Velmi častou podobou tohoto problému je i automatizované lámání hesel (při interaktivní komunikaci) pomocí slovníkových útoků nebo útoků hrubou silou.

Společným jmenovatelem je neautentizovaný či velmi špatně autentizovaný přístup uživatelů(=lidí), v rámci kterého nejsme schopni rozlišit, zda je přístup k danému prostředku realizován na žádost (pod dohledem) člověka, či zcela automatizovaně strojem. I když je teoreticky možné všechny veřejné prostředky skrýt za ochranný val silné autentizace, jde ve většině případů o krajně nevhodné řešení. Navíc se tímto způsobem nezbavíme útoku na autentizační mechanismy (lámání hesel). A blokování přístupu rozhodně pak také není všelékem.

O něco výhodnějším by byla možnost jednoduše rozhodnout, zda-li máme co do činění s člověkem, či pouze se strojem. A v případě, že jde o stroj, takový požadavek odmítnout.

Turingovy testy

Přesně pro účely rozhodování, zda jde o člověka či stroj, je možné využít veřejné Turingovy testy (dále též *CAPTCHA*, což je zkratka anglického „Completely Automated Public Turing-Test to Tell Computers and Humans Apart“ [1]). CAPTCHA je jakýkoliv algoritmicky jednoduše generovatelný test, jehož úspěšné vyřešení je jednoduché pro většinu lidí, ale nemožné (nebo velmi složité) pro stroj. Vhodnými základy testu jsou pak především takové problémy, s jejichž algoritmizací (a následně pak ideálně strojovým řešením) se lidstvo potýká již řadu let, či ještě lépe desetiletí.

V současné době se pro konstrukci Turingových testů prakticky využívají tři různé problémy:

1. rozpoznávání deformovaného textu vepsaného do obrázku,
2. rozpoznávání mluvené řeči a
3. řešení jednoduchých matematických úloh.

Ale jsou samozřejmě i další přístupy, viz např. [2].

Při návrhu CAPTCHA je nutné mít na zřeteli několik kritérií, neboť nedocenění libovolného z nich má za následek buď nepoužitelnost daného testu v praxi nebo nedostatečnou bezpečnost. Zaprvé je nutné uvažovat *rozpoznatelnost člověkem*. Je velmi jednoduché zkonstruovat test, který bude nečitelný strojem, je však neméně důležité, aby byl výsledek zároveň dostatečně jednoduše srozumitelný pro člověka.

Druhým kritériem z pohledu uživatele je dostatečná *univerzalita* testu. Test by měl být splnitelný bez ohledu na národnost jednotlivých uživatelů, proto je např. nevhodné použití akcentovaných znaků nebo azbuky, případně kanji.

Dále je dobré myslet i na *uživatele s různými smyslovými vadami* a při konstrukci vizuálního Turingova testu navíc poskytovat například audio nebo textovou variantu. Ne vždy lze ovšem všechna omezení a potřeby zohlednit...

Z pohledu vývojáře jsou klíčovými prvky i bezpečnost, technologie a náročnost na prostředky. Nezřídka se na Internetu objevují informace o CAPTCHA testech, které je možné prolomit na základě chyb v implementaci podpůrných mechanismů. Pravděpodobně nejčastější implementační chybou poslední doby je opomenutí zneplatnit test po jeho úspěšném splnění, čehož lze využít k opakování požadavků s použitím stejného identifikátoru a odpovědi. Další velmi častou chybou je použití relativně malého slovníku – pokud útočící program odhadne počet písmen a uhádne např. 80 % písmen, vyjde mu velmi málo možných řešení – často jen jedno, případně dvě.

Je tedy vhodné zamyslet se při tvorbě CAPTCHA nad opačným problémem – jak by se dala daná úloha řešit algoritmicky a co můžeme udělat pro to, abychom takovou „slabinu“ CAPCTHA odstranili. Další informace o lámání CAPTCHA, stejně jako příklady některých zranitelných obrázkových testů je možné nalézt na stránkách OCR Research Team [3].

Využití Turingových testů

Jak již bylo zmíněno v úvodu, je využití CAPTCHA poměrně široké. Nejčastěji jsou tyto testy nasazovány jako ochrana proti automatizované registraci, zejména na freemailových službách (Yahoo, Centrum, Seznam). Dále je velmi populární jejich nasazení při vstupu do autentizované oblasti, jakožto ochrana proti lámání hesel. Tento způsob ochrany autentizačních mechanismů je též jedním z výsledků diplomové práce prvního z autorů – jde o úpravu „HTTP basic“ autentizace, která před samotným ověřením uživatelského jména a hesla vyžaduje splnění Turingova testu.

Zajímavým nasazením tohoto systému je též ochrana proti nevyžádané poště (spamu) u amerického poskytovatele internetu Earthlink. Zde je nutné pro doručení dopisu do schránky příjemce, aby adresa odesílatele byla buď již povolena a nebo odesílatel splnil Turingův test, na který je mu zaslán odkaz v rámci reakce na jím odeslanou zprávu. Velmi originálním nasazením CAPTCHA je též ochrana proti distribuovaným útokům odepření služby (dále též DDoS), kterou publikoval Srikanth Kandula z MIT pod názvem Killbots [4]. Jde o modifikaci jádra operačního systému Linux, která na základě reakce

klientů na žádost o vyřešení Turingova testu dokáže účinně odstínit požadavky jednotlivých útočících strojů od požadavků legitimních klientů.

Způsoby obcházení CAPTCHA

Turingovy testy však nejsou samospasitelný mechanismus, který by nás mohl natrvalo zbavit všech problémů. Jednak mívají jednotlivé implementace, jak již bylo výše zmíněno, různé chyby a pak také existuje několik jiných způsobů, jak se dají tyto testy obejít.

Prvním způsobem je najmutí levné pracovní síly někde v rozvojové zemi, která bude zadané testy řešit v rámci své pracovní doby. Běžně by se dalo brát řešení jednoho testu za 2 vteřiny, takže průměrný dělník by mohl za hodinu vyřešit přibližně 1800 testů. To při platu 6 USD za hodinu znamená 300 testů za jeden USD. Tento způsob je nevhodný na útoky hrubou silou, ale v některých případech může jistě najít své uplatnění.

Druhým, daleko populárnějším, způsobem je využití běžných uživatelů hledajících na internetu „lechtivé obrázky“ a jiné atraktivní služby. Útočník nabídne uživatelům přístup k takovému obsahu po splnění Turingova testu. Namísto svého testu však uživateli odešle test služby, která je jeho cílem. Vzhledem k tomu, že jsou tyto testy minimálním nepohodlím, je jednoduché přesvědčit uživatele pod takovou záminkou test vyřešit a získat tak levně mnoho řešení.

Posledním způsobem je využití umělé inteligence, nebo nějakého algoritmu k řešení dané instance testu. Vzhledem k aktivnímu (člověk by až řekl bujarému) výzkumu [5] v této oblasti je možné, že se v blízké době dočkáme prolomení většiny dnes využívaných schémat. Naposledy byl totiž tímto způsobem prolomen původně složitě vypadající CAPTCHA algoritmus ez-Gimpy [6] a to s šokující úspěšností 92 %!

Výhledy do budoucnosti

Jaké jsou tedy možnosti do budoucna? Nabízí se několik možností, a ačkoli ani jedna z nich není bez nedostatků, rozhodně jsou vhodné minimálně pro zamyšlení.

První možností je využití synonym v přirozeném jazyce pro generování Turingových testů. Je poměrně jednoduché algoritmicky tvořit věty, které se liší pouze jedním slovem a žádat uživatele o výběr věty, která dává logický smysl. Díky jemným rozdílům ve významu je v dnešní době nemožné odhalit strojem význam věty a tudíž vybrat správnou alternativu. Podobným přístupem je i využití humoru [7], namísto synonym. V obou případech je na místě podotknout, že toto schéma porušuje výše zmíněnou univerzalitu testu, protože lidé, kteří mají jiný rodný jazyk, mohou mít netriviální problém takový test splnit.

Druhou možností je zavedení zvláštní formy „plateb“ za použití dané služby. Jedním z takových případů je Hashcash [8]. V tomto schématu odesílatel e-mailu do hlaviček vloží důkaz, že provedl netriviální výpočet při odesílání. Platbou je zde vlastně jistý počet procesorových cyklů vynaložený na daný výpočet. Je zřejmé, že spammer si takový výpočet pro každý e-mail nemůže dovolit, alespoň ne bez výrazného zvýšení nákladů na

jeden odeslaný dopis. Na podobném principu funguje i návrh poštovních známek [9, 10], kdy odesílatel e-mailu může ke svému e-mailu přidat elektronickou obdobu klasické poštovní známky a příjemce se může rozhodnout, zda e-mail bez známky pocházející z neplatné adresy přímo odmítne. Tímto způsobem je výrazně zvýšena ekonomická zátěž jednotlivých subjektů masivně rozesílajících spam, což povede (při patřičném rozšíření daného schématu) ke snížení množství odesílaných zpráv.

Závěr

V článku, který vznikl na základě diplomové práce [11] byly představeny veřejné Turingovy testy jako metoda ochrany veřejně přístupných služeb proti zátěži generované bez interakce člověka, pouze pomocí automatizovaných nástrojů. Přínos Turingových testů je v boji proti různým druhům spamu, ochraně autentizačních mechanismů proti hádání hesel slovníkovou metodou a v neposlední řadě i v ochraně proti distribuovaným útokům odepření služby (DDoS). Pro smysluplné využití CAPTCHA jsme představili i vlastnosti, které je nutné zohlednit při vlastní implementaci a také možné nevýhody nasazení, stejně jako jsme i přiblížili možný vývoj do budoucna.

Otevřenou otázkou, řešenou ovšem již několika týmy, je pak kvantifikace úrovně bezpečnosti získané určitými CAPTCHA mechanismy. Je totiž třeba mít na mysli skutečnost, že zatímco pro zajištění lepší úrovně bezpečnosti se často uchylujeme k používání CAPTCHA, tak další výzkumné týmy (rozeznávání obrazu a umělá inteligence v širším slova smyslu) se naopak snaží hledáním nových algoritmů strojům jejich rozhodování co nejvíce ulehčit. Nabízí se tudíž cesta kvantifikace úrovně bezpečnosti CAPTCHA právě složitostí použitých problémů z oblasti umělé inteligence [12]. Tento přístup se zatím jeví jako velmi slibný jak pro vlastní použití v oblasti bezpečnosti, tak i právě pro stimulaci rozvoje nových přístupů pro řešení daných problémů.

Michal Šafránek

Vašek Matyáš

Literatura

[1] L. von Ahn a kol. Telling Humans and Computers Apart Automatically. Communications of the ACM. Únor 2004, sv. 47, č. 2.
<http://www.cs.cmu.edu/~biglou/captcha_cacm.pdf>
[2] Carnegie Mellon University, The CAPTCHA Project.
<http://www.captcha.net/captchas/>
[3] OCR Research Team. Weak CAPTCHAs. <http://www.ocr-research.org.ua/list.html>
[4] S. Kandula a kol. Botz-4-sale: Surviving Organized DDoS Attacks That Mimic Flash Crowds. Technická zpráva TR-969, MIT, 2004.
<http://nms.lcs.mit.edu/~kandula/data/killbots.pdf>
[5] G. Mori, J. Malik. Recognizing objects in adversarial clutter – breaking a visual captcha. Computer Vision and Pattern Recognition, 2003.
<http://citeseer.ist.psu.edu/mori03recognizing.html>

[6] G. Mori. Breaking a Visual CAPTCHA.
<http://www.cs.sfu.ca/~mori/research/gimpy/>
[7] P. Ximenes a kol. A Proposal of Human Interactive Proof in the Text Domain.
<http://ximenes.info/artigos/text_only_captcha_SBSeg2005.pdf>
[8] Wikipedia. Hashcash. 2006. <http://en.wikipedia.org/wiki/Hashcash>
[9] Microsoft Research. The Penny Black Project
<http://research.microsoft.com/research/sv/PennyBlack/>
[10] S. Godin. More on Stamps. 2006.
<http://sethgodin.typepad.com/seths_blog/2006/02/more_on_stamps.html>
[11] M. Šafránek. Ochrana webových serverů proti vybraným útokům. DP FI MU Brno, 2006. <http://wejn.org/dp/>
[12] L. von Ahn a kol. CAPTCHA: Using Hard AI Problems For Security
<http://www.cs.cmu.edu/~hopper/captcha.pdf>