Kryptológia

Martin Stanek

Cieľom tohto dokumentu je poskytnúť pragmatický pohľad na kryptológiu, s dôrazom na používané kryptografické konštrukcie a ich súvis s bezpečnostnými požiadavkami. Napriek tomu, že detaily a vlastnosti kryptografických konštrukcií majú primárne matematickú povahu, obmedzíme túto stránku výkladu na minimum, aj za cenu niektorých zjednodušení. Záujemcom o hlbší pohľad na problematiku možno odporučiť špecializovanú odbornú literatúru.

Kryptológia ako vedná oblasť zahŕňa kryptografiu a kryptoanalýzu. Kryptografia sa venuje návrhu bezpečnostných konštrukcií (vo forme algoritmov, protokolov a schém) s cieľom zabezpečiť ochranu bezpečnostných atribútov dát. Kryptoanalýza skúma možnosti útokov na kryptografické konštrukcie.

1 Základné pojmy, kryptografické konštrukcie a ich ciele

V tejto časti popisujeme základné kryptografické konštrukcie zabezpečujúce dôvernosť, integritu a autentickosť (prípadne aj nepopierateľnosť autorstva) údajov. Zo základných bezpečnostných atribútov vynecháme dostupnosť, ktorú kryptografia samotná zabezpečiť nedokáže. Dostupnosť je otázkou vhodne zvolenej redundancie dát, komponentov a komunikačných pripojení, v súčinnosti s ďalšími technickými riešeniami a prevádzkovými postupmi.

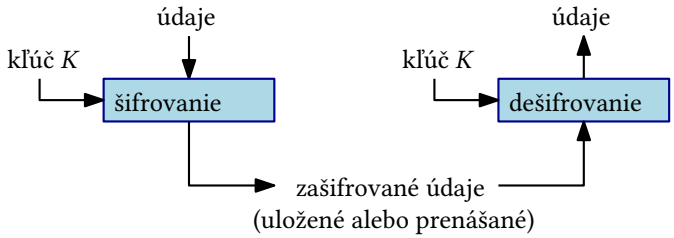
1.1 Šifrovanie

Šifrovanie slúži na zabezpečenie dôvernosti údajov. Detaily konkrétneho riešenia, akým spôsobom je šifrovanie použité, sa obvykle líšia podľa toho, či sú šifrované údaje uložené na nosiči dát (napr. disky, pásky) alebo sú prenášané počítačovými sieťami. Šifrovanie transformuje údaje pomocou šifrovacieho algoritmu a šifrovacieho kľúča do ich šifrovanej/zašifrovanej podoby. Opačný postup, teda získanie pôvodných dát z ich zašifrovanej podoby sa nazýva dešifrovanie a využíva sa pri ňom dešifrovací algoritmus a dešifrovací kľúč.

Symetrické šifrovanie

V prípade, že šifrovací a dešifrovací kľúč sú rovnaké, hovoríme o symetrických šifrách (pozri obr. 1).

Verzia 2a (september 2019)



Obr. 1: Symetrické šifrovanie

Z hľadiska bezpečnosti pri symetrickom šifrovaní očakávame, že útočník bez kľúča nie je schopný zo zašifrovaných údajov získať ich pôvodnú podobu napriek tomu, že pozná šifrovací algoritmus. V súčasnosti používaných šifrách je kľúč vyberaný ako náhodná postupnosť bitov pevnej dĺžky.

Najznámejším a najpoužívanejším symetrickým šifrovacím algoritmom je v súčasnosti AES (Advanced Encryption Standard). AES má tri varianty, líšiace sa okrem iného aj dĺžkou použitého kľúča: AES-128, AES-192, AES-256. Názov AES-n označuje variant s dĺžkou kľúča n bitov.

Dĺžka kľúča je dôležitým parametrom pre bezpečnosť šifrovacieho algoritmu – ovplyvňuje počet potenciálnych kľúčov, ktoré musí útočník vyskúšať v prípade, že sa rozhodne prezrieť priestor všetkých kľúčov. Takýto útok úplným preberaním je možné realizovať vždy, bez ohľadu na šifrovací algoritmus. Preto má počet potenciálnych kľúčov znemožňovať efektívne vyskúšanie všetkých kľúčov. V súčasnosti možno považovať kľúče s dĺžkou 128 bitov (teda 2^{128} potenciálnych kľúčov) za dostatočne bezpečné, pokiaľ nie sú slabiny v samotnom šifrovacom algoritme alebo v spôsobe generovania, distribúcie a ochrany použitých kľúčov.

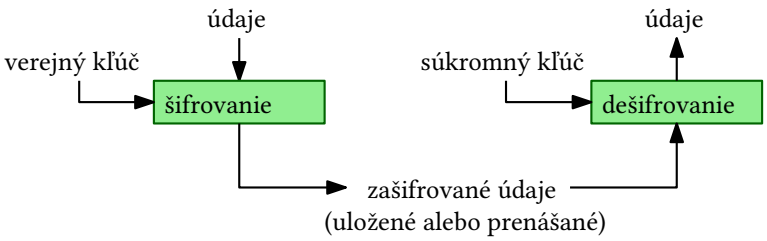
Z hľadiska efektívnosti sú symetrické šifrovacie algoritmy dostatočne rýchle na transparentné šifrovanie a dešifrovanie diskov osobných počítačov, komunikácie v počítačových sieťach a podobne, pričom spomalenie spôsobené takýmto dodatočným spracovaním údajov je zanedbateľné. Viaceré hardvérové zariadenia sú v súčasnosti konštruované so zabudovanou podporou pre kryptografické operácie, napríklad novšie procesory obsahujú podporu špeciálnych inštrukcií pre implementáciu AES.

Asymetrické šifrovanie

Samostatná trieda šifrovacích algoritmov využíva na šifrovanie iný kľúč ako na dešifrovanie, pozri obr. 2, pričom dešifrovací kľúč nie je možné efektívne vypočítať zo šifrovacieho kľúča. V tomto prípade hovoríme o asymetrickom šifrovaní, prípadne o šifrovaní s verejným kľúčom. Ako názov napovedá, šifrovací kľúč, bežne označovaný ako verejný kľúč, je zvyčajne zverejnený a teda ktokoľvek môže šifrovať. Dešifrovať je možné len so znalosťou dešifrovacieho kľúča, ten je obvykle označovaný ako súkromný kľúč. Najznámejším príkladom asymetrického šifrovania je RSA schéma.

Asymetrické šifry sú konštruované s využitím vhodných matematických problémov, napr. rozklad (faktorizácia) veľkých čísel na súčin prvočiniteľov. Svoju bezpečnosť opierajú o zložitosť riešenia týchto problémov. Kľúče v asymetrických šifrách preto reprezentujú konkrétne matematické objekty (a nie sú to náhodne volené postupnosti bitov). Pri rovnakej miere kryptografickej odolnosti šifry je dĺžka kľúčov asymetrických šifier zvyčajne podstatne dlhšia ako dĺžka kľúčov symetrickej šifry. Napríklad dĺžka RSA kľúčov 3072 bitov poskytuje rovnakú mieru kryptografickej odolnosti ako AES-128 [7].

Z hľadiska bezpečnosti asymetrického šifrovania očakávame, že útočník nie je schopný bez znalosti



Obr. 2: Asymetrické šifrovanie

súkromného kľúča zo zašifrovaných údajov získať ich pôvodnú podobu (alebo nejakú netriviálnu informáciu o pôvodných údajoch). Pripomeňme, že šifrovací kľúč je verejne známy, a teda útočník má možnosť zašifrovať ľubovoľné údaje.

Hybridné šifrovanie

Asymetrické šifrovanie a dešifrovanie sú z hľadiska výpočtových nárokov oveľa náročnejšie ako ich symetrické náprotivky. Sú vhodné najmä na šifrovanie krátkych údajov, tými sú v praxi najčastejšie symetrické kľúče v tzv. hybridných šifrovacích schémach. Hybridná šifrovacia schéma kombinuje symetrický a asymetrický šifrovací algoritmus nasledujúcim spôsobom (pozri obr. 3):

Šifrovanie – odosielateľ

Vstup: dáta M, verejný kľúč príjemcu
Posielané údaje (výstup): EK, EM

1. odosielateľ vygeneruje náhodný symetrický kľúč K
2. zašifruje údaje M symetrickou šifrou s použitím kľúča K (výsledok označme EM)
3. zašifruje kľúč K asymetrickým šifrovaním s použitím verejného kľúča príjemcu (výsledok označme EK)

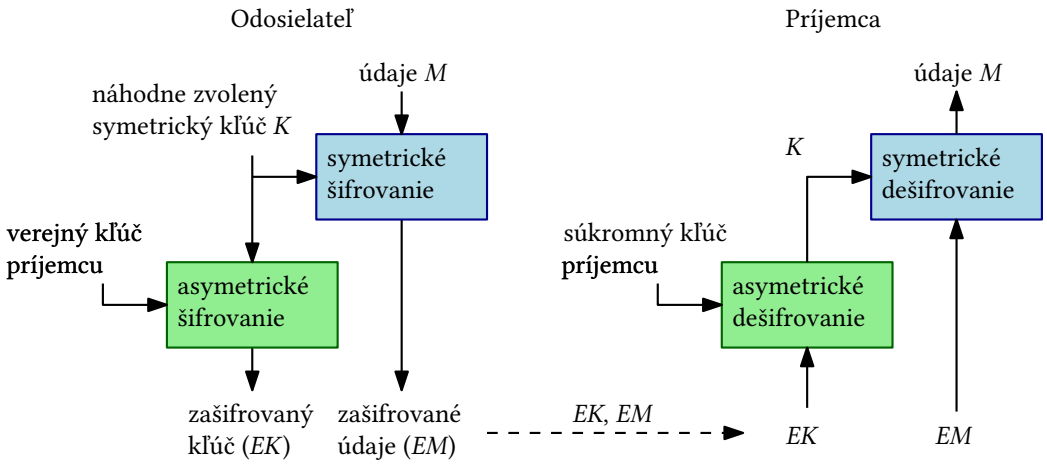
Dešifrovanie – príjemca

Vstup: EK, EM, vlastný súkromný kľúč
Výstup: M

1. príjemca získa K dešifrovaním EK, pričom použije svoj súkromný kľúč
2. získa pôvodné dáta dešifrovaním EM, pričom použije kľúč K

Popis predpokladá posielanie údajov odosielateľom k príjemcovi, avšak hybridný prístup môže byť použitý aj v prípade, že šifrovanie a dešifrovanie vykonáva rovnaká osoba (vlastník súkromného kľúča) a dáta sú ukladané lokálne, napr. na disk.

Výhodou hybridného prístupu je efektívne šifrovanie, keď rozsahom veľké údaje sú šifrované rýchlym symetrickým algoritmom, pričom bezpečnú distribúciu symetrického kľúča rieši asymetrické šifrovanie. Pred použitím takejto schémy stačí dôveryhodným spôsobom distribuovať verejného kľúča príjemcu. Obvykle tento problém rieši infraštruktúra verejných kľúčov (pozri časť 3.3). Verejný kľúč je nemenný dlhší čas, napr. jeden rok, a môže byť používaný opakovane.



Obr. 3: Hybridné šifrovanie

Porovnanie a použitie

Typické rozdiely medzi symetrickým a asymetrickým šifrovaním sumarizuje nasledujúca tabuľka:

	Symetrické šifrovanie	Asymetrické šifrovanie
Primárne použitie	dôvernosť údajov ľubovoľnej veľkosti	dôvernosť krátkych dát (typicky napr. kľúče pre symetrické šifrovanie)
Komunikácia	1:1 – obvykle dvaja účastníci (jeden odosielateľ, jeden príjemca)	N:1 – ľubovoľný počet odosielateľov (šifrovací kľúč je verejný), jeden príjemca (súkromný dešifrovací kľúč)
Efektívnosť	rýchle šifrovanie aj dešifrovanie	pomalé šifrovanie aj dešifrovanie
Dĺžka kľúčov	obvykle 128 až 256 bitov (náhodný reťazec bitov)	v závislosti na konkrétnom algoritme, niekoľko sto až niekoľko tisíc bitov
Distribúcia kľúčov	obvykle potrebné použiť kryptografické protokoly na distribúciu (dohodnutie) kľúča	relatívne jednoduchá distribúcia verejného kľúča (avšak potrebné overiť jeho autentickosť)

Šifrovací algoritmus, či už symetrický alebo asymetrický, neposkytuje ochranu integrity ani autentickosti prenášaných údajov. Teda skutočnosť, že údaje boli prenášané/uložené zašifrované a úspešne sme ich dešifrovali neznamená, že počas prenosu/uloženia zašifrované dáta neboli útočníkom zmenené. Výnimkou sú špecifické konštrukcie módov symetrických šifier, tzv. autentizované šifrovanie. V súčasnosti sú v praxi používané čoraz častejšie a príkladmi takýchto módov sú GCM (Galois/Counter Mode) a CCM (Counter with CBC-MAC). Použitie AES v týchto módoch býva potom označené ako AES-GCM, resp. AES-CCM. Pokiaľ nevyužívame autentizované šifrovanie, je potrebné na zabezpečenie integrity a autentickosti údajov použiť iné kryptografické konštrukcie, najčastejšie autentizačné kódy správ (pozri časť 1.2).

Šifrovanie možno nájsť v praxi vo veľkom počte rôznorodých aplikácií, pričom pre symetrické šifrovanie sa štandardne využíva AES (v módoch vhodných pre daný účel). Uveďme niekoľko príkladov:

- Šifrovanie diskov osobných počítačov, kde sa údaje transparentne pri čítaní z disku dešifrujú a pri zápise na disk šifrujú – Bitlocker (štandardný nástroj v operačnom systéme Windows), VeraCrypt (multiplatformová aplikácia), FileVault 2 (štandardný nástroj v macOS). Cieľom takýchto riešení je znížiť riziko prezradenia údajov, napr. pri odcudzení prenosného počítača.

- Šifrovanie komprimovaných archívov (napr. zip) – viaceré aplikácie pre prácu s komprimovanými archívmi údajov umožňujú okrem komprimácie vzniknuté archívy aj zašifrovať s použitím symetrického šifrovania (napr. 7-Zip, WinZip používajú AES). Šifrovací kľúč je vypočítaný zo zadaného hesla. Zašifrovaný archív je následne možné dešifrovať a rozbaliť len s použitím tohto hesla. V prípade ad-hoc potreby poslať citlivé údaje, pričom nemáme k dispozícii verejný kľúč príjemcu (alebo tento ani žiadny verejný kľúč nemá), je často najjednoduchším riešením údaje zabaliť do šifrovaného archívu s použitím dostatočne silného hesla. Následne archív pošleme príjemcovi mailom a heslo oznámime iným komunikačným kanálom (povedzme SMS). Samozrejme, pokiaľ útočník získa zašifrovaný archív aj prenášané heslo, dokáže dešifrovať rovnako ako príjemca.

- Šifrovanie komunikácie v nezabezpečených sieťach, napr. na internete. V súčasnosti je prezeranie väčšiny web stránok zabezpečené protokolom TLS (Transport Layer Security). Jeho použitie je signalizované v adrese stránky prostredníctvom „https://“ namiesto „http://“, ako aj vizuálnym indikátorom, najčastejšie v podobe zámku. TLS okrem iných atribútov zabezpečuje aj dôvernosť prenášaných údajov symetrickým šifrovaním, pričom konkrétny použitý algoritmus sa dohodne pri nadviazaní spojenia medzi internetovým prehliadačom a webovým serverom.

1.2 Hašovacie funkcie a autentizačné kódy správ

Hašovacie funkcie

Kryptografické hašovacie funkcie sú algoritmy, ktoré z prakticky ľubovoľne dlhého vstupu vypočítajú hodnotu – reťazec bitov pevnej dĺžky (ten nazveme odtlačok). Ide o deterministické algoritmy, teda pre rovnaký vstup je vypočítaný vždy rovnaký odtlačok. Úlohou odtlačku je jednoznačne reprezentovať vstupné údaje/dokument. Pre bežne používané hašovacie funkcie má odtlačok dĺžku 256 bitov v prípade SHA-256 alebo 512 bitov v prípade SHA-512. V niektorých konštrukciách a starších protokoloch sa možno stretnúť aj s hašovacími funkciami SHA-1 resp. MD5 (s odtlačkami dĺžky 160 resp. 128 bitov). Z hľadiska rýchlosti spracovania vstupu sú hašovacie funkcie porovnateľné so symetrickými šifrovacími algoritmami.

Primárne použitie hašovacích funkcií je v ďalších kryptografických konštrukciách, napríklad v autentizačných kódoch správ, schémach digitálnych podpisov a pod. Hašovacie funkcie nevyužívajú žiadny kľúč a teda ktokoľvek vie vypočítať odtlačok k ľubovoľnému vstupu. Preto má samostatné použitie hašovacích funkcií význam len pre detekciu narušenia integrity údajov pri náhodnej (necielenej) zmene, alebo v situáciách, keď útočník nemá úplnú kontrolu nad všetkými komunikačnými kanálmi a nemôže okrem údajov modifikovať aj ich odtlačok. V opačnom prípade útočník ľahko dopočíta korektný odtlačok k pozmeneným údajom. Uveďme dva ilustračné príklady použitia hašovacích funkcií na kontrolu integrity:

- Distribúcia objemných súborov (softvér, video a pod.) na internete, kde je na webovej stránke zverejnený odtlačok takéhoto súboru. Po stiahnutí súboru môže používateľ lokálne vypočítať jeho odtlačok a porovnať hodnotu s odtlačkom zverejneným na internete. Nesúlad vypočítaného a zverejneného odtlačku signalizuje, že pri prenose údajov došlo k modifikácii, napríklad spôsobenej nespoľahlivým prenosom alebo zámernou úpravou. Samozrejme, pokiaľ útočník dokáže zmeniť pri prenose nielen údaje samotné, ale aj informáciu o odtlačku z webovej stránky, používateľ nič podozrivé nespozoruje. Zdôraznime, že hašovacie funkcie vo všeobecnosti nezabezpečujú autentickosť údajov.
- Ochrana integrity súborov vypočítaním ich odtlačkov. Pokiaľ odtlačky odložíme (napr. na neprepisovateľné médium), dokážeme neskôr opätovným výpočtom odtlačkov a ich porovnaním s odloženými hodnotami zistiť, či a ktorý zo súborov bol modifikovaný. Keďže odtlačky sú obvykle podstatne kratšie ako zdrojové súbory, tento spôsob ochrany poskytuje len detekciu narušenia integrity a neumožňuje rekonštruovať pôvodný obsah súborov (na tento účel slúži zálohovanie). Na druhej strane je porovnávanie odtlačkov prevádzkovo jednoduchšie ako porovnávanie celých kópií súborov.

Pri prezentácii odtlačkov v čitateľnej forme je obvykle použitý zápis v šestnástkovej (hexadecimálnej) sústave využívajúcej cifry `0`, `1`, …, `9`, `A`, `B`, `C`, `D`, `E`, `F`. Príklady odtlačkov niekoľkých reťazcov pomocou SHA-256:

vstupný reťazec	SHA-256
Kryptologia	`928926C86A99D03F47FECA0A85F52298D9D7F2AEEB31CAFEC7B76198DB9A8E61`
kryptologia	`4978DBF2A4D3B02ABA050EFAB96B86CB844C18E9DDF4044B047BF32DF9914B99`

Použitie hašovacích funkcií v kryptografických konštrukciách vyžaduje, aby hašovacie funkcie mali vhodné bezpečnostné vlastnosti. Dve základné vlastnosti sú odolnosť vzoru a odolnosť voči kolíziám:

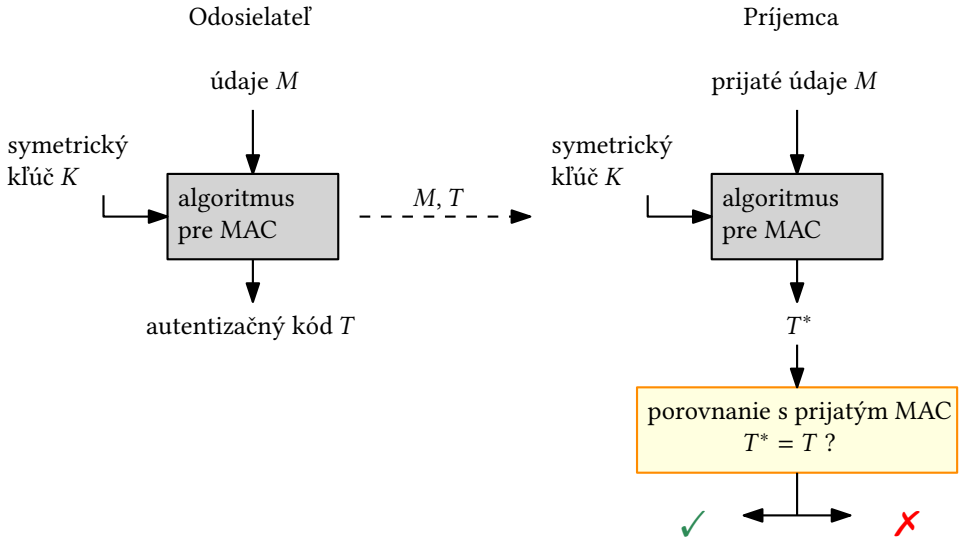
- Odolnosť vzoru: k danému odtlačku nie je efektívne možné vypočítať vstup s takýmto odtlačkom.
- Odolnosť voči kolíziám: nie je efektívne možné vypočítať dva rôzne vstupy s rovnakým odtlačkom.

Autentizačné kódy správ

Autentizačný kód správy (angl. Message Authentication Code, skrátene MAC) je v podstate odtlačok správy, pri výpočte ktorého bol použitý kľúč. Algoritmy pre výpočet autentizačných kódov správ sú akoby hašovacie funkcie s kľúčom, pričom sú často konštruované práve z hašovacích funkcií. Najznámejšou konštrukciou je HMAC – ide o všeobecnú konštrukciu, kde konkrétny algoritmus dostaneme voľbou „podkladovej“ hašovacej funkcie (napr. HMAC-SHA1 je HMAC skonštruovaný z hašovacej funkcie SHA-1).

Keďže výpočet odtlačku závisí na kľúči, autentizačné kódy správ zabezpečujú autentickosť údajov. Samozrejme, iba v prípade ak je kľúč známy len oprávneným používateľom. Najčastejšie použitie

autentizačných kódov je pri ochrane komunikácie v počítačovej sieti (pozri obr. 4). V takom prípade kľúč poznajú spoločne odosielateľ a príjemca. Pri posielaní údajov k nim odosielateľ pripojí autentizačný kód. Príjemca vypočíta z prijatých údajov a kľúča autentizačný kód a porovná ho s prijatým autentizačným kódom. V prípade zhody je potvrdená autentickosť údajov. Pokiaľ útočník nepozná kľúč použitý pri výpočte odtlačku, nedokáže údaje nepozorovane modifikovať, lebo nevie dopočítať správny autentizačný kód.



Obr. 4: Použitie autentizačných kódov správ

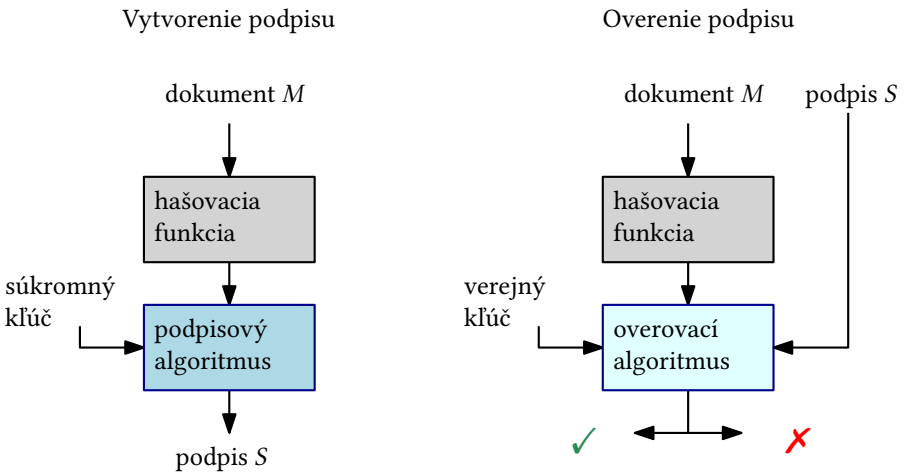
Dohodnutie/distribúcia konkrétneho kľúča na takýto účel je zvyčajne úlohou vhodného kryptografického protokolu pri nadväzovaní spojenia (napríklad niektoré varianty TLS protokolu alebo IKE protokol v rámci IPsec). Následne je výpočtom autentizačného kódu zabezpečený každý prenášaný paket – algoritmy pre MAC sú dostatočne rýchle (porovnateľne so symetrickým šifrovaním).

Autentizačné kódy nezabezpečujú nepopierateľnosť autorstva prenášaných správ. Keďže príjemca má k dispozícii rovnaký kľúč ako odosielateľ, autentizačný kód ľubovoľnej správy vie vypočítať sám. To znamená, že odosielateľ dokáže poprieť autorstvo správy.

1.3 Digitálne podpisy

Schémy pre digitálne podpisy sú asymetrické kryptografické konštrukcie, pozostávajúce z podpisového algoritmu a z overovacieho algoritmu. Používateľ vytvorí inštanciu schémy vygenerovaním dvojice kľúčov – súkromného a verejného. Podpisový algoritmus vytvára digitálny podpis z dokumentu a zo súkromného kľúča. Overovací algoritmus overuje korektnosť konkrétneho podpisu na základe dokumentu a verejného kľúča. Verejný kľúč je obvykle zverejnený a teda podpis môže overiť ktokoľvek.

Z dôvodu efektívnosti aj z bezpečnostných dôvodov nie je fakticky podpisovaný dokument ako taký, ale jeho odtlačok vypočítaný zvolenou hašovacou funkciou (pozri obr. 5). Preto je dôležité, aby hašovacia funkcia spĺňala vlastnosti spomínané v časti 1.2. Napríklad možnosť nájsť kolízie v hašovacej funkcii (teda dva rôzne dokumenty M_1, M_2 s rovnakým odtlačkom) znamená, že podpis dokumentu M_1 je zároveň korektným podpisom dokumentu M_2.



Obr. 5: Schéma pre digitálne podpisy

Najznámejšími schémami pre digitálne podpisy sú RSA a DSA (Digital Signature Algorithm, niekedy vo variante ECDSA využívajúcom tzv. eliptické krivky). Poznamenajme, že štandardná RSA podpisová schéma sa od RSA schémy pre asymetrické šifrovanie líši vo viacerých implementačných detailoch. Napriek tomu je matematická povaha asymetrického páru kľúčov rovnaká a niekedy je jeden pár kľúčov používaný na oba účely (teda v schéme pre asymetrické šifrovanie aj v schéme pre digitálne podpisy), hoci sa to neodporúča.

Napriek tomu, že verejný kľúč aj podpisový a overovací algoritmus sú známe, nie je efektívne možné bez znalosti súkromného kľúča vytvoriť k ľubovoľnému dokumentu korektný podpis. To znamená, že digitálne podpisy poskytujú ochranu integrity a autentickosti údajov. Navyše, ak je používateľ jediný, kto pozná svoj súkromný kľúč, tak korektný podpis dokumentu znemožňuje používateľovi poprieť vlastný podpis (hovoríme o nepopierateľnosti autorstva). Samozrejme, praktické použitie digitálnych podpisov si vyžaduje vyriešiť dôveryhodnú distribúciu verejných kľúčov, možnosť vyhlásiť neplatnosť verejného kľúča po prípadnom prezradení súkromného kľúča a množstvo ďalších praktických otázok. Tie sa snaží riešiť tzv. infraštruktúra verejných kľúčov (pozri časť 3.3) aj s príslušným právnym rámcom[1].

Tabuľka 1 porovnáva základné charakteristiky hašovacích funkcií, autentizačných kódov a digitálnych podpisov.

2 Protokoly

Kryptografické protokoly sú postupnosť krokov a výmen správ medzi dvoma alebo viacerými účastníkmi, s cieľom naplniť konkrétne bezpečnostné požiadavky. Pritom protokoly využívajú rôzne kryptografické konštrukcie. Pokiaľ spracovanie údajov zahŕňa interakciu a prenos údajov medzi systémami alebo používateľmi, je z hľadiska bezpečnosti typicky potrebné zabezpečiť dve základné požiadavky:

1. Autentizácia komunikujúcich účastníkov – každý účastník si vie overiť, že komunikuje so

[1] V prostredí EÚ je to Nariadenie Európskeho parlamentu a Rady (EÚ) č. 910/2014 o elektronickej identifikácii a dôveryhodných službách pre elektronické transakcie na vnútornom trhu (známe aj ako eIDAS) a súvisiace politiky a štandardy.

	Hašovacie funkcie	Autentizačné kódy	Digitálne podpisy
Integrita	áno	áno	áno
Autentickosť	nie	áno	áno
Nepopierateľnosť autorstva	nie	nie	áno
Kľúče	žiadne	symetrické	asymetrický pár kľúčov
Efektívnosť	rýchle	rýchle	pomalé
Typická aplikácia	kontrola integrity statických dát	autentickosť paketov pri prenose v sieti	autentickosť dokumentov

Tabuľka 1: Porovnanie hašovacích funkcií, autentizačných kódov a digitálnych podpisov

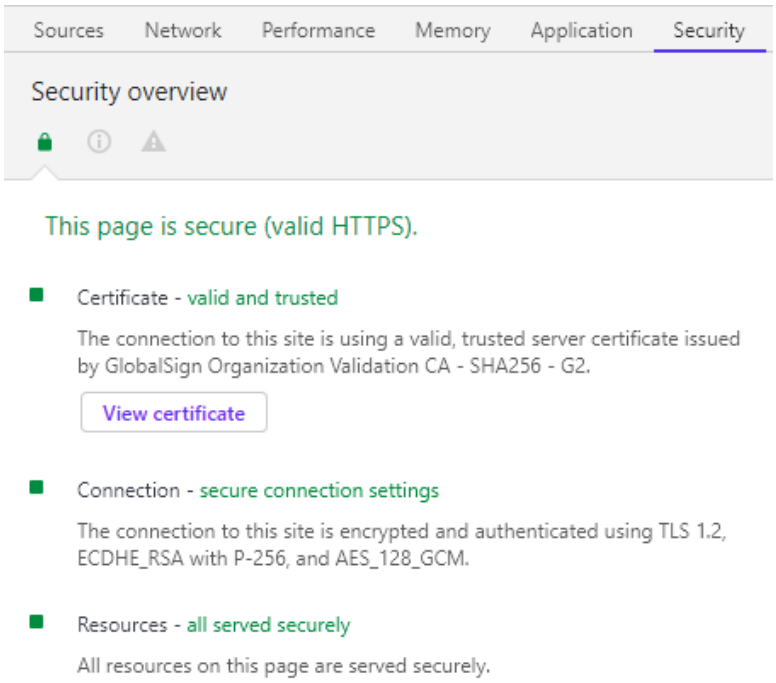
želaným partnerom.

2. Dohodnúť a distribuovať kryptografické kľúče (a ďalšie parametre), ktoré sa v následnej komunikácii použijú na šifrovanie, výpočet autentizačných kódov, prípadne na zabezpečenie iných bezpečnostných atribútov pomocou vhodných kryptografických konštrukcií.

Uvedené požiadavky napĺňa najvýznamnejšia trieda kryptografických protokolov – protokoly pre autentizáciu a dohodnutie kľúčov. Prirodzene, tieto protokoly sú obvykle vykonané pri nadväzovaní spojenia. Najpoužívanejším protokolom tohto typu je TLS (Transport Layer Security), niekedy sa možno stretnúť aj so starším označením SSL (Secure Sockets Layer). Ďalším príkladom je IKE (Internet Key Exchange), využívaný v rámci štandardu IPsec. Účastníka možno autentizovať len na základe nejakých dôveryhodne distribuovaných informácií. Takou informáciou môže byť verejný kľúč, napr. vo forme certifikátu (najčastejší spôsob v prípade autentizácie webového servera v TLS) alebo spoločná tajná informácia dohodnutá a distribuovaná dôveryhodným spôsobom vopred (pomerne častý spôsob v prípade IPsec). V prípade verejného kľúča dokazuje účastník svoju identitu tým, že preukáže znalosť prislúchajúceho súkromného kľúča – typicky podpíše vhodnú správu využijúc svoj súkromný kľúč, alebo je schopný dešifrovať dáta šifrované s použitím jeho verejného kľúča.

Keďže používateľ sa pri používaní webu stretne s TLS, ilustrujme použitie kryptografických techník práve na tomto protokole. V TLS je použitý komunikačný model klient-server, pričom klient je ten účastník protokolu, ktorý zahajuje komunikáciu, napr. webový prehliadač používateľa. V úvode protokolu si klient a server dohodnú sadu nimi preferovaných a podporovaných kryptografických techník. TLS ponúka istú flexibilitu pri voľbe algoritmov a metód dohodnutia kryptografických kľúčov, skúsme sa preto zamerať na jednu z možností, pričom sa opäť nevyhneme značnému zjednodušovaniu. Server pošle klientovi svoj verejný kľúč vo forme certifikátu podpísaného nejakou certifikačnou autoritou. Pokiaľ má klient vhodným spôsobom získaný verejný kľúč certifikačnej autority a dôveruje jej, dokáže overiť digitálny podpis na certifikáte a tým autentickosť verejného kľúča servera. Server použije svoj súkromný kľúč na podpísanie údajov slúžiacich pre dohodnutie kľúčov, ktoré spolu s popisom pošle klientovi. Následne klient prostredníctvom verejného kľúča servera overí autentickosť získaných údajov. Klient vytvorí vlastné údaje, ktoré spoločne s údajmi servera umožnia odvodiť kľúče a ďalšie potrebné parametre. Zároveň tieto údaje pošle aj serveru, ktorý vykoná rovnaké odvodenie.

Po získaní kľúčov je nadviazanie spojenia dokončené a vytvorený zabezpečený komunikačný kanál je k dispozícii aplikácii (teda napr. webovému prehliadaču).



Obr. 6: Parametre TLS na stránke Európskej komisie, https://ec.europa.eu/ (jún 2019)

Obrázok 6 ilustruje informácie o TLS spojení, ktoré vytvorí prehliadač s webovým serverom Európskej komisie. Z uvedeného si možno všimnúť verziu TLS protokolu (1.2), šifrovací algoritmus (AES s kľúčom dĺžky 128 bitov), použitý mód (GCM zabezpečujúci autentizované šifrovanie) a mechanizmus použitý na dohodnutie kľúčov (ECDHE_RSA využívajúci eliptickú krivku P-256). Zároveň je vidieť, že certifikát verejného kľúča webového servera vydala certifikačná autorita GlobalSign, pričom ďalšie informácie o certifikáte sú k dispozícii samostatne.

Základné charakteristiky TLS sú zhrnuté v tabuľke 2. Poznamenajme, že konkrétne techniky podstatne závisia na verzii TLS ako aj na konfigurácii klienta a servera. Napríklad vo verzii TLS 1.3, štandardizovanej v roku 2018, sú na šifrovanie prípustné výlučne konštrukcie poskytujúce autentizované šifrovanie.

3 Heslá a kryptografické kľúče

3.1 Heslá

Heslá sú najpoužívanejším autentizačným prostriedkom. Sú príkladom autentizácie založenej na znalosti, na rozdiel od autentizačných mechanizmov založených na vlastníctve (niečo čo máte, typicky rôzne hardvérové tokeny) alebo identite (niečo čím ste, typicky biometrické metódy). Heslo je reťazec znakov a obvykle ho volí používateľ sám. V niektorých systémoch/aplikáciách je obmedzená dĺžka ako aj abeceda hesla – napr. v prípade PIN kódov používaných pri platobných kartách, pri SIM kartách v mobilných telefónoch, alebo v prípade bezpečnostného osobného kódu (BOK) k eID karte.

	TLS (v závislosti na verzii a konfigurácii)
Autentizácia servera	povinná (znalosť súkromného kľúča k verejnému kľúču uvedenom v certifikáte)
Autentizácia klienta	voliteľná (málokedy používané, obvykle riešené po vytvorení TLS spojenia samostatne)
Distribúcia kľúčov	viaceré protokoly (odvodenie kľúčov a ďalších parametrov pre šifrovanie, prípadne pre autentizačné kódy)
Dôvernosť	symetrické šifrovanie (podpora rôznych algoritmov a módov)
Autentickosť	autentizačné kódy alebo autentizované šifrovanie (podpora rôznych algoritmov)
Úprava aplikácie využívajúcej protokol	zvyčajne potrebné v aplikácii špecificky inicializovať komunikačný kanál

Tabuľka 2: Základné charakteristiky TLS

Na bezpečnosť autentizácie využívajúcej heslá vplýva viacero faktorov, uveďme tie najvýznamnejšie:

- Dĺžka a „náhodnosť“ hesla – čím je heslo dlhšie a náhodnejšie, tým viac pokusov potrebuje útočník na jeho uhádnutie. Pri heslách volených používateľmi je dostatočná náhodnosť problematická (aj schopnosť pamätať si náhodné heslá). V praxi je preto zvyčajne požiadavka na dĺžku hesla dôležitejšia ako jeho náhodnosť.
- Spôsob prenosu a overovania hesla – heslo má byť prenášané cez komunikačný kanál so zabezpečenou dôvernosťou, ako prevencia pred odpočutím hesla útočníkom.
- Spôsob uloženia hesla na strane používateľa – v ideálnom prípade si používateľ heslá pamätá, používa rôzne heslá v rôznych systémoch a nemá spoločné heslá s inými používateľmi. Vhodnou alternatívou je použitie aplikácií na správu hesiel.
- Spôsob uloženia hesla na strane systému (servera) – heslá nie sú uložené v otvorenom tvare, pre zníženie dopadov kompromitácie servera (ak útočník získa neoprávnený prístup k údajom servera).
- Ďalšie parametre autentizácie – definovanie počtu neúspešných pokusov zadania hesla, po ktorom sa prístup používateľa zablokuje (v prípade PIN kódov obvykle 3), vynútenie zmeny iniciálneho hesla a iné opatrenia znižujúce pravdepodobnosť úspešného útoku.

Ďalší spôsob použitia hesiel je odvodenie symetrických kryptografických kľúčov. Predstavme si situáciu, že používateľ má svoj súkromný kľúč pre podpisovú schému uložený v súbore na lokálnom disku. Pre minimalizáciu rizika prezradenia kľúča je tento súbor zašifrovaný (obvykle aj s nejakou formou ochrany integrity). Keďže používateľ si pravdepodobne nie je schopný zapamätať povedzme 128 bitov dlhý, náhodne zvolený symetrický kľúč, tento sa v podobných situáciách odvodí z hesla. Teda z hesla zvoleného používateľom je vypočítaný symetrický šifrovací kľúč a ten následne použitý

na šifrovanie alebo dešifrovanie súboru so súkromným kľúčom. Podobne sa kľúče z hesiel odvádzajú aj v iných situáciách. Samozrejme, náhodnosť takto získaného kľúča je nižšia ako keby bol volený skutočne náhodne. Na druhej strane je používateľ schopný si heslo zapamätať. Bezpečnosť takéhoto použitia hesiel je ovplyvnená aj konkrétnym algoritmom, ktorým sa heslo transformuje na kľúč. Podobne ako pre iné kryptografické konštrukcie, aj v tomto prípade existujú vhodné štandardy (napr. PBKDF2 – Password-Based Key Derivation Function 2 [6]).

Koľko bitov náhodnosti sa skrýva v hesle? Porovnanie odhadovanej náhodnosti hesiel volených používateľom voči dĺžke náhodného symetrického kľúča nie je priamočiare. Schopnosť útočníka hádať heslo závisí na tom, či je heslo alebo jeho podstatná časť zo slovníka, rôznorodosti znakov, použití číselných postupností, opakovaní znakov, dĺžke a pod. Silu zvoleného hesla odhadujú niektoré webové stránky pri vytvorení účtu, programy na správu hesiel alebo špecializované aplikácie. Na ilustráciu uvádzame v nasledujúcej tabuľke porovnanie odhadov sily niekoľkých hesiel prostredníctvom programu na správu hesiel KeePass[2] a knižnice zxcvbn[3] určenej na odhad sily hesiel. Čísla v tabuľke vyjadrujú ekvivalentnú dĺžku symetrického kľúča v bitoch.

heslo	KeePass	zxcvbn
`qwerty`	12	2.32
`password1`	8	7.57
`JE38bslk@psl`	67	39.86
`spidersarecoolandfun`	72	50.66

O sile používateľských hesiel je možné urobiť si predstavu z útokov v reálnom prostredí. V roku 2012 bola publikovaná databáza odtlačkov hesiel približne 6,5 milióna používateľov služby LinkedIn. Jednoduchý slovníkový útok bez špeciálneho hardvéru umožnil v priebehu 4 hodín zistiť heslá cca. 900 tisíc používateľov. Ďalšie pokračovanie v slovníkovom útoku viedlo celkovo k takmer 2 miliónom zistených hesiel. Nezanedbateľná časť používateľov volí a používa pomerne slabé heslá. Podobný záver možno spraviť aj z databáz uniknutých hesiel rôznych webových služieb. Pätnásť najčastejších hesiel v takýchto databázach v roku 2018 sú v nasledujúcej tabuľke (pričom autori analýzy[4] uvádzajú, že takmer 10% používateľov použilo niektoré heslo z top 25):

1.	123456	6.	111111	11.	princess
2.	password	7.	1234567	12.	admin
3.	123456789	8.	sunshine	13.	welcome
4.	12345678	9.	qwerty	14.	666666
5.	12345	10.	iloveyou	15.	abc123

3.2 Kľúče

Spôsob narábania s kryptografickými kľúčmi je najvýznamnejším faktorom ovplyvňujúcim bezpečnosť kryptografických konštrukcií v praxi. Správa kryptografických kľúčov zahŕňa hlavne nasledujúce činnosti:

[2] https://keepass.info/ (august 2019)

[3] https://github.com/dropbox/zxcvbn (august 2019)

[4] SplashData's Top 100 Worst Passwords of 2018; štatistika na základe viac ako 5 miliónov uniknutých hesiel, najmä používateľov zo Severnej Ameriky a západnej Európy.

1. Generovanie kľúčov – postupy vytvárania kľúčov, vrátane použitých zdrojov náhodnosti. Kľúče musia byť nepredikovateľné, vyberané z dostatočne veľkej množiny.
2. Distribúcia kľúčov – spôsob doručenia kľúčov používateľom alebo na cieľový systém/server, vrátane naplnenia bezpečnostných požiadaviek pri distribúcii kľúčov (napr. dôvernosť, autentickosť).
3. Ukladanie a prístup ku kľúčom – spôsob uloženia kľúčov a opatrenia pre riadenie prístupu k nim nielen počas prevádzky, ale aj pri zálohovaní a archivácii.
4. Ničenie kľúčov – spôsob vymazania kľúčov, ktoré už nie sú potrebné.

Pri správe kľúčov je vhodné definovať aj postupy pri kompromitácii alebo pri zneplatňovaní kľúčov, intervaly výmen kľúčov a pod.

V prípade adekvátnej správy kľúčov je bezpečnosť kryptografických konštrukcií určená hlavne ich kryptografickou kvalitou a dĺžkou používaných kľúčov. Inštitúcie ako NIST (National Institute of Standards and Technology), NSA (National Security Agency), BSI (nemecký Bundesamt für Sicherheit in der Informationstechnik) a iné vydávajú odporučenia pre vhodné algoritmy a dĺžky kľúčov. Napríklad NSA zverejnila požiadavky na dĺžky kľúčov a algoritmy v tzv. Commercial National Security Algorithm (CNSA) Suite [1], pre kryptografickú ochranu údajov klasifikovaných až po TOP SECRET. Pre zabezpečenie dôvernosti je požadovaný AES-256, pre digitálne podpisy RSA schéma s aspoň 3072 bitov dlhým verejným kľúčom alebo ECDSA schéma konkrétnou štandardizovanou eliptickou krivkou, atď.

Z povahy kryptografických konštrukcií vyplýva, že útočník môže hľadať symetrický kľúč vyskúšaním všetky možností. Podobne môže riešiť konkrétny matematický problém v prípade asymetrických šifrovacích schém alebo schém pre digitálne podpisy. Ilustrujme schopnosť útočníka prezrieť redukovaný priestor kľúčov pre algoritmus AES-128 v nasledujúcej tabuľke. Súčasné procesory sú schopné s hardvérovou podporou AES realizovať niekoľko sto miliónov dešifrovacích operácií AES-128 za sekundu (využijúc všetky jadrá). Predpokladajme 300 miliónov operácií za sekundu. V stĺpcoch je uvažovaný individuálny útočník s jedným počítačom, a stredne veľká firma s 500 počítačmi. Hodnoty v tabuľke vyjadrujú veľkosť priestoru, ktorý dokáže útočník za daný čas prezrieť, pričom veľkosť priestoru je vyjadrená dĺžkou symetrického kľúča v bitoch (napr. 40 bitov je $2^{40} = 1\,099\,511\,627\,776$ možností).

Čas útoku	Individuálny útočník (1 procesor)	Stredne veľká firma (500 procesorov)
1 minúta	34	43
1 hodina	40	49
1 deň	45	54
30 dní	49	58
1 rok	53	62
100 rokov	60	69

Tabuľka má výlučne ilustratívny charakter, iné algoritmy majú inú rýchlosť, procesory sa postupne zrýchľujú a zlacňujú, špecializované zákaznícke integrované obvody skonštruované na takýto účel

majú v pomere k cene vyšší výkon. Napríklad 1024 krát rýchlejší procesor by znamenal pripočítanie 10 k hodnotám uvedeným v tabuľke. Bez ohľadu tieto fakty, ponúka tabuľka dobrú predstavu o exponenciálnom raste počtu potenciálnych kľúčov s rastom ich dĺžky a snáď aj o dostatočnej dĺžke 128 alebo 256 bitového kľúča.

Je dôležité si uvedomiť, že tabuľka ukazuje možnosti útočníka v situácii, keď sú kľúče volené skutočne náhodne a s rovnakou pravdepodobnosťou. Útočník je v oveľa lepšej situácii, ak sú niektoré kľúče pravdepodobnejšie ako iné, prípadne ak sa niektoré kľúče určite nepoužijú. To platí napríklad v situácii, ak sú kľúče odvodené z hesiel.

Viaceré kryptografické konštrukcie, napr. niektoré podpisové schémy a protokoly, využívajú ďalšie náhodne volené parametre. Náhodnosť, prípadne dôvernosť týchto parametrov má priamy dopad na bezpečnosť konštrukcií. Ilustratívnym príkladom je implementácia digitálnych podpisov v herných konzolách PlayStation 3 spoločnosti Sony, s cieľom zabrániť nahratiu a spusteniu neautorizovaného (nepodpísaného) kódu. Použitie statického namiesto náhodného parametra pri podpisovaní algoritmom ECDSA viedlo v roku 2010 k prezradeniu súkromného podpisového kľúča a útočník následne dokázal podpísať akýkoľvek kód.

3.3 Infraštruktúra verejných kľúčov

Asymetrické kryptografické konštrukcie, či už pre šifrovanie alebo digitálne podpisy, majú výhodu v tom, že verejné kľúče môžu byť zverejnené a teda nie je potrebné zabezpečovať ich dôvernosť. Problémom je však autentickosť verejných kľúčov. Ako sa odosielateľ údajov uistí o tom, že verejný šifrovací kľúč patrí naozaj adresátovi, a teda nepošle údaje šifrované verejným kľúčom, ktorý podstrčil útočník? Ako sa vieme presvedčiť, že verejný kľúč, ktorý použijeme na overenie digitálneho podpisu, skutočne patrí konkrétnemu autorovi dokumentu?

Tento problém je ľahko riešiteľný v prostredí s malým počtom účastníkov, ktorí sa navzájom poznajú a môžu si svoje verejné kľúče odovzdať osobne. Takéto riešenie sa však nedá aplikovať vo všeobecnom prípade veľkého počtu vzdialených alebo vopred neznámych účastníkov. Cieľom infraštruktúry verejných kľúčov je definovať technické (kryptografické) a organizačné metódy a postupy na dosiahnutie dôveryhodnej distribúcie verejných kľúčov. Infraštruktúra verejných kľúčov (PKI – public key infrastructure) prenáša dôveru účastníkov na dôveryhodný subjekt – certifikačnú autoritu. Certifikačná autorita vydáva certifikáty verejných kľúčov, čo sú digitálne podpísané dátové štruktúry obsahujúce, okrem ďalších atribútov (podrobnejší príklad je uvedený v prílohe):

- jednoznačnú identifikáciu subjektu, pre ktorý je certifikát vydaný, napríklad doménové meno web servera, meno a e-mailová adresa používateľa a pod.,
- verejný kľúč subjektu, vrátane identifikácie konkrétneho kryptografického algoritmu, pre ktorý je kľúč určený,
- účel použitia verejného kľúča, či slúži na šifrovanie alebo overovanie podpisov (tým zároveň hovorí o účele použitia zodpovedajúceho súkromného kľúča),
- interval platnosti certifikátu, určujúci odkedy a dokedy je certifikát platný,
- digitálny podpis certifikačnej autority, umožňujúci overiť autentickosť všetkých údajov v certifikáte.

Fungovanie PKI sa opiera o dva predpoklady. Prvým je dôvera účastníkov, že certifikačná autorita si plní svoje úlohy čestne a bezpečne. Druhým je dôveryhodná distribúcia verejného kľúča certifikačnej autority, pomocou ktorého si vedia účastníci overiť podpis certifikačnej autority v certifikátoch a teda autentickosť údajov v nich obsiahnutých. Obvykle sa takáto distribúcia vykoná zároveň s distribúciou softvéru, napr. webové prehliadače obsahujú po inštalácii zoznam niekoľkých desiatok certifikátov verejných kľúčov certifikačných autorít (tzv. koreňových certifikátov), ktorým používatelia implicitne dôverujú. V organizáciách sa často certifikáty certifikačných autorít, napr. vlastných, používaných na interné účely, distribuujú prostredníctvom nástrojov na správu koncových zariadení (PC, notebooky, mobilné telefóny). Verejný kľúč certifikačnej autority sa distribuuje vo formáte „samopodpísaného" certifikátu, teda podpis je možné overiť verejným kľúčom, ktorý je uvedený v certifikáte. Samozrejme, samopodpísaný certifikát s ľubovoľnou sadou atribútov si vie vygenerovať ktokoľvek, preto je dôležitý mechanizmus, akým sa certifikáty certifikačných autorít distribuujú.

Hlavnou úlohou certifikačnej autority je vydávať certifikáty verejných kľúčov. To vyžaduje overiť identitu subjektu, ktorý žiada o certifikát, aby nemohol útočník žiadať certifikát napríklad pre server accounts.google.com alebo www.paypal.com. Zároveň certifikačná autorita overuje ďalšie skutočnosti, napr. znalosť súkromného kľúča zodpovedajúceho k verejnému kľúču subjektu. Činnosti certifikačnej autority, ktoré nevyžadujú prácu so súkromným kľúčom sú často delegované na registračnú autoritu, zabezpečujúcu kontakt so zákazníkmi (teda subjektmi so záujmom o vydanie vlastného certifikátu). Sú známe príklady, keď bezpečnostné zlyhania certifikačnej autority alebo jej registračnej autority viedli k vydaniu falošných certifikátov. Holandská certifikačná autorita DigiNotar v dôsledku bezpečnostných problémov vyhlásila v roku 2011 bankrot. V rovnakom roku zápasila s kompromitáciou používateľského účtu v registračnej autorite certifikačná autorita Comodo. Pochybná prax pri vydávaní certifikátov certifikačnými autoritami prevádzkovanými spoločnosťou Symantec (Thawte, GeoTrust a ďalšie) viedla v rokoch 2017 a 2018 postupne až k tomu, že webové prehliadače prestali dôverovať certifikátom vydaným týmito certifikačnými autoritami pred decembrom 2017. Aktivity Symantecu v tejto oblasti prevzala spoločnosť DigiCert.

Infraštruktúra verejných kľúčov musí byť pripravená aj na situáciu, keď bude súkromný kľúč niektorého subjektu prezradený. V takom prípade je potrebné zneplatniť certifikát ešte pred uplynutím intervalu platnosti uvedeného v certifikáte. V princípe sa používajú dve riešenia:

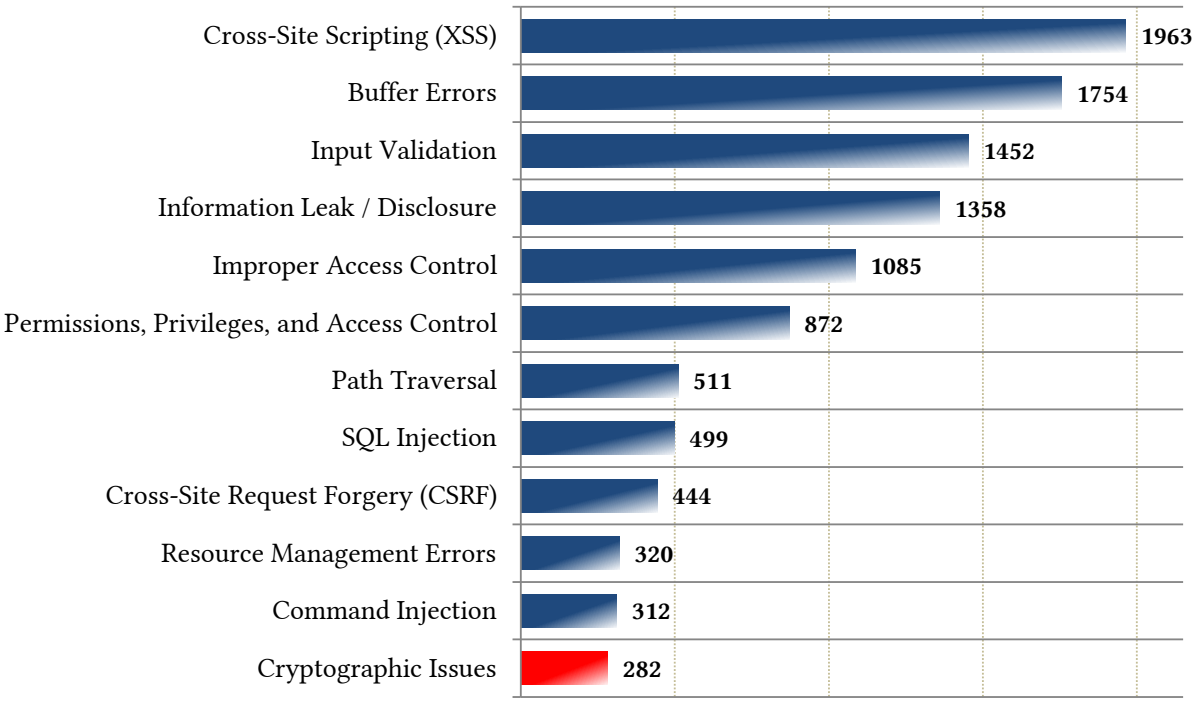
- zoznam zneplatnených certifikátov (CRL, Certificate revocation list) – periodicky vydávaný zoznam podpísaný certifikačnou autoritou, obsahujúci sériové čísla zneplatnených certifikátov;
- interaktívny protokol (OCSP, Online Certificate Status Protocol), ktorý umožňuje spýtať sa certifikačnej autority na platnosť konkrétneho certifikátu online.

4 Zraniteľnosti a kryptografia

Najznámejšou databázou softvérových zraniteľností je NVD (National Vulnerability Database), ktorú prevádzkuje NIST. NVD zraniteľnosti klasifikuje podľa typu, závažnosti a iných atribútov. V roku 2018 bolo v NVD publikovaných viac ako 16 500 zraniteľností[5]. Graf na obr. 7 zobrazuje početnosť 12 najčastejších typov zraniteľností publikovaných v roku 2018. Samozrejme, samotný počet zraniteľností nehovorí nič o ich závažnosti alebo reálnej zneužiteľnosti v praxi. Nakoniec, stačí jedna zraniteľnosť na kompromitáciu práve vášho systému, servera, siete alebo aplikácie. Samostatnú kategóriu *Cryptographic*

[5]Zdroj: https://nvd.nist.gov/home.cfm (jún 2019)

issues tvoria problémy spojené s implementáciou, použitím alebo naopak absenciou kryptografie, ktoré v roku 2018 tvorili cca. 1,7% zraniteľností.



Obr. 7: Top-12 kategórií zraniteľností publikovaných v NVD v roku 2018

Štatistiku zneprehľadňuje zavedenie viacerých samostatných, detailnejších kategórií, takže niektoré zraniteľnosti sú klasifikované v týchto kategóriách (a nie v Cryptographic issues). Počty zraniteľností pre vybrané ďalšie kategórie súvisiace s kryptografiou uvádzame v nasledujúcej tabuľke:

Kategória	Počet (rok 2018)
Use of Hard-coded Credentials	124
Improper Certificate Validation	112
Key Management Errors	38
Inadequate Encryption Strength	35
Improper Verification of Cryptographic Signature	30
Use of Cryptographically Weak Pseudo-Random Number Generator	13
Use of a Broken or Risky Cryptographic Algorithm	11
Insufficient Entropy	4

Častým problémom je absencia kryptografických opatrení ako takých – napríklad prenos citlivých údajov alebo aktualizácia softvéru cez nezabezpečené spojenie. Predchádzajúca tabuľka poukazuje aj na ďalšie problémy súvisiace s implementáciou kryptografie:

- použitie fixných hesiel alebo kľúčov pre servisné účty alebo takéto heslá/kľúče odvodené z verejne známych údajov,

- nedostatočná (neúplná) kontrola certifikátov alebo podpisov,
- chyby pre správe kľúčov,
- použitie neadekvátnych kryptografických algoritmov (slabé šifrovacie algoritmy alebo algoritmy s nedostatočnou dĺžkou kľúča, slabé generátory pseudonáhodných čísel a pod.), atď.

Iný pohľad na zraniteľnosti súvisiace s kryptografiou ponúka napr. správa spoločnosti CA Veracode [9], kde na základe výsledkov testovania softvéru pomocou statickej analýzy kódu boli Cryptographic issues druhou najčastejšie sa vyskytujúcou kategóriou zraniteľností (v prípade dynamického testovania kódu treťou).

5 Štandardy a legislatívne požiadavky

Väčšina v praxi používaných kryptografických konštrukcií je štandardizovaná. Šifrovacie algoritmy (symetrické aj asymetrické schémy), hašovacie funkcie, autentizačné kódy, schémy pre digitálne podpisy, protokoly, ako aj ďalšie konštrukcie sú k dispozícii vo forme štandardov. To zvyšuje vzájomnú interoperabilitu a čiastočne predchádza bezpečnostným zraniteľnostiam a chybám vďaka otvorenej možnosti ich analyzovať a pripomienkovať. Najčastejšie používané štandardy vydáva NIST a z pochopiteľných dôvodov sú široko akceptované výrobcami softvéru a hardvérových zariadení. Kryptografické protokoly, napr. TLS, IPSec, SSH a podobne, sú najčastejšie štandardizované vo forme RFC (Request for Comments).

Štandardy v oblasti informačnej bezpečnosti sa venujú kryptografii skôr okrajovo, pričom sa sústredia najmä na správu kľúčov a používanie štandardných kryptografických konštrukcií. Medzinárodný štandard ISO/IEC 27001:2013 [4] definuje pre systém riadenia informačnej bezpečnosti nasledujúce požiadavky v oblasti kryptografie (vhodné naplnie požiadaviek prostredníctvom opatrení možno potom nájsť v ISO/IEC 27002:2013 [5]):

- Politika používania kryptografických opatrení na ochranu informácií – zahŕňa vytvorenie a implementáciu príslušnej politiky.
- Správa kľúčov – zahŕňa vytvorenie a implementáciu politiky týkajúcej sa používania a ochrany kryptografických kľúčov počas ich životného cyklu.

Bezpečnostné hodnotenie a certifikácia IT systémov je cieľom Spoločných kritérií na hodnotenie bezpečnosti informačných technológií[6], známych ako „Common Criteria“. Konkrétna verzia bola vydaná aj ako štandard ISO/IEC 15408. Podľa Spoločných kritérií je možné hodnotiť rôzne komponenty ako sú napríklad operačný systém, čipová karta, firewall, databázový server, smerovač a pod. Zoznam certifikovaných produktov je dostupný na stránke projektu. Z hľadiska kryptografie definujú Spoločné kritériá funkčnú triedu *Kryptografická podpora* s dvoma množinami požiadaviek:

- Správa kryptografických kľúčov – zahŕňa všeobecné požiadavky na generovanie, distribúciu, prístup a deštrukciu kľúčov s tým, že v systéme sú používané štandardizované konštrukcie.

[6] https://www.commoncriteriaportal.org/ (jún 2019)

- Prevádzka kryptografie – všeobecná požiadavka na vykonávanie kryptografických operácií v súlade s explicitne definovanými štandardami.

Štandard FIPS 140-2 vydal NIST a definuje bezpečnostné požiadavky pre kryptografické moduly. Ide o najčastejšie používaný štandard pre bezpečnostné posúdenie kryptografických modulov. Moduly môžu byť rôznorodé – kryptografická knižnica operačného systému, šifrovaný pamäťový USB kľúč, čipová karta, hardvérový bezpečnostný modul a pod. Štandard definuje 4 bezpečnostné úrovne, od úrovne 1 až po úroveň 4 s postupne sprísňovanými požiadavkami. Medzi oblasti, v ktorých sú požiadavky definované patria špecifikácia modulu, role, služby, autentizácia, fyzická bezpečnosť modulu, samotestovanie, správa kľúčov, elektromagnetické vyžarovanie a ďalšie. Počet vydaných osvedčení pre jednotlivé úrovne v roku 2018 je uvedený v nasledujúcej tabuľke. Pre zaujímavosť uveďme, že k júnu 2019 existuje celkovo len 15 certifikátov modulov na úrovni 4 podľa FIPS 140-2 (z čoho sú 4 platné, ostatné sú historické).

Úroveň podľa FIPS 140-2	2018
Level 1	159
Level 2	72
Level 3	25
Level 4	1

NIST vydal v roku 2019 novú verziu štandardu FIPS 140-3 [8]. Ten vychádza zo štandardov ISO/IEC 19790:2012 [2] a ISO/IEC 24759:2017 [3], v ktorých niektoré časti modifikuje a spresňuje vlastnými požiadavkami. Testovanie kryptografických modulov podľa FIPS 140-3 má začať v roku 2020 a testovanie podľa FIPS 140-2 má byť ukončené v roku 2021.

Podotknime, že certifikácia konkrétneho produktu nie je zárukou jeho bezpečnosti. Certifikácia je overenie splnenia konkrétnych požiadaviek a nie bezpečnostná analýza. Ilustratívnym príkladom boli šifrované pamäťové USB kľúče spoločností Verbatim, Kingstone a SanDisk, certifikované na úrovni 2 podľa FIPS 140-2. V roku 2010 sa ukázalo, že k dešifrovaniu a získaniu prístupu k údajom postačuje jednoduchá úprava riadiaceho programu bez znalosti prístupového kľúča. Ďalším príkladom je DUHK (Don't Use Hard-coded Keys) útok, publikovaný v roku 2017, ktorý využíva použitie schváleného (do roku 2016), avšak zastaraného generátora náhodných čísel spolu s pevne nastavenou iniciálnou hodnotou. Dôsledkom takejto kombinácie je schopnosť útočníka predikovať výstup generátora, čo v niektorých prípadoch znamená získanie kryptografických kľúčov. Zaujímavosťou je, že k identifikácii zraniteľných certifikovaných zariadení prispeli práve dokumenty z certifikačného procesu FIPS 140-1 a 140-2.

Napriek uvedenému majú certifikáty produktov svoj význam – hovoria o tom, že tvorcovia museli naplniť isté bezpečnostné požiadavky. Produkt s vhodnou úrovňou certifikácie podľa Spoločných kritérií a FIPS 140-2 vzbudzuje väčšiu dôveru ako produkt bez certifikácie.

5.1 Legislatíva SR

Niektoré kryptografické požiadavky možno nájsť aj v normatívnych právnych aktoch SR. V tejto časti uvedieme vybrané príklady. Poznamenajme, že požiadavky obvykle s istým oneskorením reflektujú realitu, dostupnosť konkrétnych technológií a sú formulované všeobecne. Záujemcom o skúmanie týchto

a ostatných zmienok o kryptografii v právnych predpisoch možno odporučiť funkciu vyhľadávania na portáli Slov-Lex[7].

Výnos Ministerstva financií SR č. 55/2014 Z. z. o štandardoch pre informačné systémy verejnej správy v znení neskorších predpisov[8] uvádza v časti Technické štandardy:

- (§3) používanie skupiny protokolov Internet Procotol Security (IPsec) na zabezpečenie sieťových protokolov;
- (§4, §6, §7, §10) podpora kryptografického protokolu Transport Layer Security (TLS) pre chránený prenos dát, pri prenose elektronických poštových správ, pri chránenom prístupe k verejným elektronickým poštovým službám;
- (§8) používanie formátu Secure/Multipurpose Internet Mail Extensions (S/MIME) pri chránenom prenose elektronických poštových správ, pri chránenom verejnom prístupe k adresárovým službám;
- (§9) používanie protokolu Hypertext Transfer Protocol (HTTP) s Transport Layer Security (TLS) na zabezpečenie prenosu dát medzi klientom a webovým serverom a medzi webovými servermi.

Pre obsah webového sídla (§15) požaduje výnos zverejnenie kontaktnej informácie, na ktorej možno získať „kontrolný reťazec znakov“ na overenie pravosti používaných certifikátov verejných kľúčov pre elektronické služby verejnej správy a elektronické poštové správy; zverejnenie najmenej jedného verejného kľúča pre chránený prenos elektronických poštových správ, ak povinná osoba takýto prenos poskytuje. Zaujímavý je aj §33 venovaný ochrane proti škodlivému kódu, kde sa okrem iného ocitli aj požiadavka na podporu zabezpečenia autenticity a integrity súborov pomocou kryptografických prostriedkov, najmä elektronického podpisu a požiadavka na podporu šifrovania elektronických dokumentov. V správe cloud computingu (§55) sa v manažmente rizík požaduje vyhodnotiť hrozby zlyhania správy kryptografických kľúčov alebo ich ich kompromitácie. Ďalšie požiadavky možno nájsť v prílohách výnosu.

Štandardizáciu a v oblasti elektronických podpisov a súvisiacej infraštruktúry potrebnú pre implementáciu nariadenia Európskeho parlamentu a Rady (EÚ) č. 910/2014 o elektronickej identifikácii a dôveryhodných službách pre elektronické transakcie na vnútornom trhu (eIDAS) má na starosti ETSI (Európsky inštitút pre telekomunikačné normy). Detaily o prípustných kryptografických algoritmoch možno nájsť v technických normách a špecifikáciách pripravovaných príslušnou technickou komisiou ETSI.

Do pôsobnosti zákona č. 215/2004 o ochrane utajovaných skutočností v znení neskorších prepisov patrí aj šifrová ochrana informácií, teda zabezpečenie ochrany utajovaných skutočností kryptografickými metódami. Keďže bezpečnostné štandardy pre oblasť šifrovej ochrany informácií (a ďalšie podrobnosti o kryptografických metódach) sú utajovanými skutočnosťami Národného bezpečnostného úradu, nie sú verejne prístupné.

[7] https://www.slov-lex.sk/ (júl 2019)
[8] https://www.slov-lex.sk/static/pdf/2014/55/ZZ_2014_55_20190601.pdf (júl 2019)

6 Praktické rady na záver

Cieľom tejto časti je ponúknuť niektoré základné praktické rady týkajúce sa výberu a použitia kryptografických konštrukcií. Odporúčania nie sú vyčerpávajúce, ide o čisto subjektívne názory autora.

- ✓ Používajte štandardné kryptografické algoritmy, schémy a protokoly. Kryptografia nie je miesto na kreativitu a ad-hoc riešenia.
- ✓ Používajte kryptografické konštrukcie na dosiahnutie tých bezpečnostných atribútov, pre ktoré sú určené. Napríklad (štandardné, „neautentizované“) šifrovanie nezabezpečuje integritu ani autentickosť údajov, autentizačné kódy ani digitálne podpisy nezabezpečujú dôvernosť údajov.
- ✓ Používajte dostatočné dĺžky kľúčov a dbajte na kvalitu (náhodnosť) generovania kľúčov.
- ✓ Pravidelne meňte kľúče. Dlhodobo nezmenené kľúče považujte za prezradené.
- ✓ Voľte dostatočne dlhé heslá. Obvykle je heslo najslabším „kľúčom“ v systéme. Voľte rozličné heslá pre rôzne systémy a zvážte použitie aplikácie pre správu hesiel.
- ✓ Majte premyslené, čo robiť po kompromitácii kľúčov alebo hesiel.
- ✓ Ak môžete, použite certifikované riešenia. Poznajte rozsah a podmienky certifikácie. Pamätajte, že certifikácia nie je náhradou bezpečného používania.
- ✓ Poznajte konfiguračné možnosti kryptografických riešení a ich bezpečnostné dopady. Preferujte nastavenia podľa best-practice odporučení v danej oblasti.
- ✓ Dôsledne overujte certifikáty verejných kľúčov – meno subjektu, certifikačná autorita, aktuálna platnosť, interval platnosti, účel použitia a pod. Samopodpísaný certifikát nehovorí nič o autentickosti verejného kľúča.
- ✓ Koreňové certifikáty certifikačných autorít získajte dôveryhodným spôsobom.
- ✓ Venujte pozornosť relevantným bezpečnostným hrozbám a rizikám. Kryptografia nenahradí iné organizačné a technické bezpečnostné opatrenia.

7 Otázky a úlohy

Riešenie jednoduchých úloh a zamyslenie sa nad vybranými otázkami súvisiacimi s používaním kryptografických techník má pomôcť k lepšiemu pochopeniu a možno aj prehĺbeniu prebraných tém. Úlohy zámerne nemajú jediné riešenie.

1. Pomocou vhodného programu vytvorte šifrovaný archív (napr. zip). Aký šifrovací algoritmus je pritom použitý? Pokúste sa rozbaliť archív s nesprávnym aj so správnym heslom.
2. Stiahnite niektorú bezpečnostnú aktualizáciu produktu spol. Microsoft[9] a overte jej SHA-256 odtlačok s tým, ktorý je publikovaný na webe.

[9] https://portal.msrc.microsoft.com/en-us/security-guidance/summary (júl 2019)

3. Nájdite stránky testujúce kvalitu hesiel a vyskúšajte kvalitu niekoľkých hesiel. Akým spôsobom je prezentovaná kvalita hesla a aké parametre o kvalite rozhodujú?

4. Pre vybranú webovú stránku pomocou prehliadača zistite, aká certifikačná autorita vydala certifikát servera, pre aký algoritmus je určený verejný kľúč a aký je interval platnosti certifikátu.

5. Pre vybranú webovú stránku pomocou prehliadača zistite, aké sú parametre TLS spojenia – verzia, použité algoritmy.

6. Konfiguráciu TLS protokolu na webových serveroch dostupných z internetu možno otestovať viacerými voľne dostupnými službami. Najznámejšou je SSL Server Test[10]. Vyskúšajte, aké hodnotenie a výstupy služba produkuje pre vami zvolené web servery.

7. Zistite, akým certifikačným autoritám dôveruje váš prehliadač. Koľko ich je?

8. Na stránke NVD nájdite niektorú zraniteľnosť aplikácie, databázového systému alebo operačného systému, ktorý používate. Všimnite si rozsah evidovaných informácií a atribútov.

9. Vyberte si niektorý kryptografický modul certifikovaný podľa FIPS 140-2. Pozrite sa na to, aké informácie o module sú uvedené v certifikáte a v súvisiacej dokumentácii (Security Policy).

10. Nájdite kryptografické algoritmy prípustné pre podpisové schémy v rámci eIDAS. Ktorá technická norma ETSI ich definuje? Aké sú požadované dĺžky kľúčov?

Literatúra

[1] *Commercial National Security Algorithm (CNSA) Suite Factsheet*. National Security Agency. 2015. URL: https://apps.nsa.gov/iaarchive/library/ia-guidance/ia-solutions-for-classified/algorithm-guidance/commercial-national-security-algorithm-suite-factsheet.cfm (cit. 07/2019) (cit. na s. 13).

[2] *ISO/IEC 19790:2012, Information technology – Security techniques – Security requirements for cryptographic modules*. International Organization for Standardization, 2012 (cit. na s. 18).

[3] *ISO/IEC 24759:2017, Information technology – Security techniques – Test requirements for cryptographic modules*. International Organization for Standardization, 2017 (cit. na s. 18).

[4] *ISO/IEC 27001:2013, Information technology – Security techniques – Information security management systems – Requirements*. International Organization for Standardization, 2013 (cit. na s. 17).

[5] *ISO/IEC 27002:2013, Information technology – Security techniques – Code of practice for Information Security Controls*. International Organization for Standardization, 2013 (cit. na s. 17).

[6] K. Moriarty, B. Kaliski a A. Rusch. *PKCS #5: Password-Based Cryptography Specification Version 2.1*. RFC 8018. 2017. URL: https://tools.ietf.org/html/rfc8018 (cit. na s. 12).

[7] *NIST Special Publication 800-57 Recommendation for Key Management – Part 1: General (Revision 4)*. National Institute of Standards a Technology, 2016. DOI: 10.6028/NIST.SP.800-57pt1r4 (cit. na s. 2).

[10] https://www.ssllabs.com/ssltest/ (júl 2019)

[8] *Security requirements for cryptographic modules*. FIPS PUB 140-3. National Institute of Standards a Technology, 2019. DOI: 10.6028/NIST.FIPS.140-3 (cit. na s. 18).

[9] *State Of Software Security*. Volume 9. CA Veracode, 2018. URL: https://www.veracode.com/state-of-software-security-report (cit. na s. 17).

Príloha: príklad štruktúry certifikátu

Uvádzame príklad štruktúry certifikátu, v tomto prípade z web stránky Európskej komisie. Na získanie a zobrazenie certifikátu sme použili program OpenSSL vo verzii 1.1.1.

```
$ openssl s_client -connect ec.europa.eu:443 </dev/null 2>/dev/null |
  openssl x509 -outform PEM >ec.pem
$ openssl x509 -in ec.pem -noout -text
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      3b:5c:f9:73:c5:1f:95:c0:91:8c:57:e5
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: C = BE, O = GlobalSign nv-sa, CN = GlobalSign Organization
            Validation CA - SHA256 - G2
    Validity
      Not Before: Apr  3 15:21:05 2018 GMT
      Not After : Jun  9 11:31:05 2020 GMT
    Subject: C = BE, ST = Brussels, L = Brussels, O = European Commission,
             CN = *.ec.europa.eu
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
        RSA Public-Key: (2048 bit)
        Modulus:
          00:e9:6a:61:2b:ab:72:90:dc:37:33:4a:55:71:bd:
          ... vynechaných 16 riadkov ...
          81:ab
        Exponent: 65537 (0x10001)
    X509v3 extensions:
      X509v3 Key Usage: critical
        Digital Signature, Key Encipherment
      Authority Information Access:
        CA Issuers - URI:http://secure.globalsign.com/cacert/
                     gsorganizationvalsha2g2r1.crt
        OCSP - URI:http://ocsp2.globalsign.com/gsorganizationvalsha2g2
      X509v3 Certificate Policies:
        Policy: 1.3.6.1.4.1.4146.1.20
        CPS: https://www.globalsign.com/repository/
        Policy: 2.23.140.1.2.2
      X509v3 Basic Constraints:
        CA:FALSE
      X509v3 CRL Distribution Points:
        Full Name:
        URI:http://crl.globalsign.com/gs/gsorganizationvalsha2g2.crl
      X509v3 Subject Alternative Name:
        DNS:*.ec.europa.eu, DNS:ec.europa.eu
```

```
    X509v3 Extended Key Usage:
      TLS Web Server Authentication, TLS Web Client Authentication
    X509v3 Subject Key Identifier:
      F1:D7:68:F9:7D:50:4F:DE:87:92:DF:C9:05:AC:B2:00:20:20:FF:F0
    X509v3 Authority Key Identifier:
      keyid:96:DE:61:F1:BD:1C:16:29:53:1C:C0:CC:7D:3B:83:00:40:E6:1A:7C
    CT Precertificate SCTs:
      Signed Certificate Timestamp:
        Version   : v1 (0x0)
        Log ID    : 87:75:BF:E7:59:7C:F8:8C:43:99:5F:BD:F3:6E:FF:56:
                    8D:47:56:36:FF:4A:B5:60:C1:B4:EA:FF:5E:A0:83:0F
        Timestamp : Apr  3 15:21:07.077 2018 GMT
        Extensions: none
        Signature : ecdsa-with-SHA256
                    30:46:02:21:00:C0:56:60:E2:55:6C:99:04:D7:33:F9:
                    ... vynechané 3 riadky ...
                    EA:17:8D:F5:41:DD:B3:8C
      Signed Certificate Timestamp:
        Version   : v1 (0x0)
        Log ID    : BB:D9:DF:BC:1F:8A:71:B5:93:94:23:97:AA:92:7B:47:
                    38:57:95:0A:AB:52:E8:1A:90:96:64:36:8E:1E:D1:85
        Timestamp : Apr  3 15:21:07.663 2018 GMT
        Extensions: none
        Signature : ecdsa-with-SHA256
                    30:46:02:21:00:B1:75:6E:35:8C:D2:F2:4C:77:CE:12:
                    ... vynechané 3 riadky ...
                    C8:E7:4B:4E:05:DB:21:D5
      Signed Certificate Timestamp:
        Version   : v1 (0x0)
        Log ID    : 6F:53:76:AC:31:F0:31:19:D8:99:00:A4:51:15:FF:77:
                    15:1C:11:D9:02:C1:00:29:06:8D:B2:08:9A:37:D9:13
        Timestamp : Apr  3 15:21:07.165 2018 GMT
        Extensions: none
        Signature : ecdsa-with-SHA256
                    30:46:02:21:00:96:58:40:9E:66:A0:D8:CD:70:79:BF:
                    ... vynechané 3 riadky ...
                    26:B1:83:8A:C3:AA:6C:E3
Signature Algorithm: sha256WithRSAEncryption
   23:59:40:d3:bd:e3:94:c6:fd:28:0a:eb:73:14:99:31:81:ab:
   ... vynechaných 13 riadkov ...
   cd:31:d0:34
```