Ověřování identity, hesla a digitální podpis

Marko Hölbl

Anotace

Tento kurz seznamuje s problematikou procesu ověřování (autentizace) založeném na systému hesel včetně popisu základních konceptů a návazností na využívané technologie. Dále je podrobněji rozpracována role digitálního podpisu v procesu ověřování, definovány základní pojmy a uvedeny používané techniky.

Cíle

Tento kurz poskytuje základní informace o problematice ověřování (autentizace), jeho základních komponentách a procesu ověřování založeném na systému hesel, a to včetně postupů, jak vhodně chránit hesla na straně uživatele i ověřovatele. Podrobněji jsou zmíněny koncepty správy hesel, oblast vícestupňové (multifázové) autentizace a autentizace bez hesla.
Dále jsou uvedeny informace o technických možnostech implementace digitálního podpisu včetně popisu hashovacích funkcí, kryptografie veřejného klíče a infrastruktury veřejného klíče. Na závěr je uveden digitální podpis aktuálně nedílná součást bezpečného procesu ověřování.

Klíčová slova

hesla, autentizace, digitální podpis, infrastruktura veřejných klíčů, hashovací funkce

Datum vytvoření

06.01.2022

Časová dotace

15 hodin

Jazyková verze

česky

Licence

ISBN

Literatura

[1] Batten, L. M. (2013). Public key cryptography: applications and attacks, John Wiley & Sons.

[2] Boonkrong, S. (2021). Authentication and Access Control: Practical Cryptography Methods and Tools, Springer.

[3] Buchmann, J., et al. (2013). Introduction to public key infrastructures, Springer.

[4] Burnett, M. (2006). Perfect password: Selection, protection, authentication, Elsevier.

[5] Grassi, P. A., et al. (2017). "NIST special publication 800-63b: digital identity guidelines." National Institute of Standards and Technology (NIST).

[6] Grimes, R. A. (2020). Hacking Multifactor Authentication, John Wiley & Sons.

KAPITOLA 1

Úvod

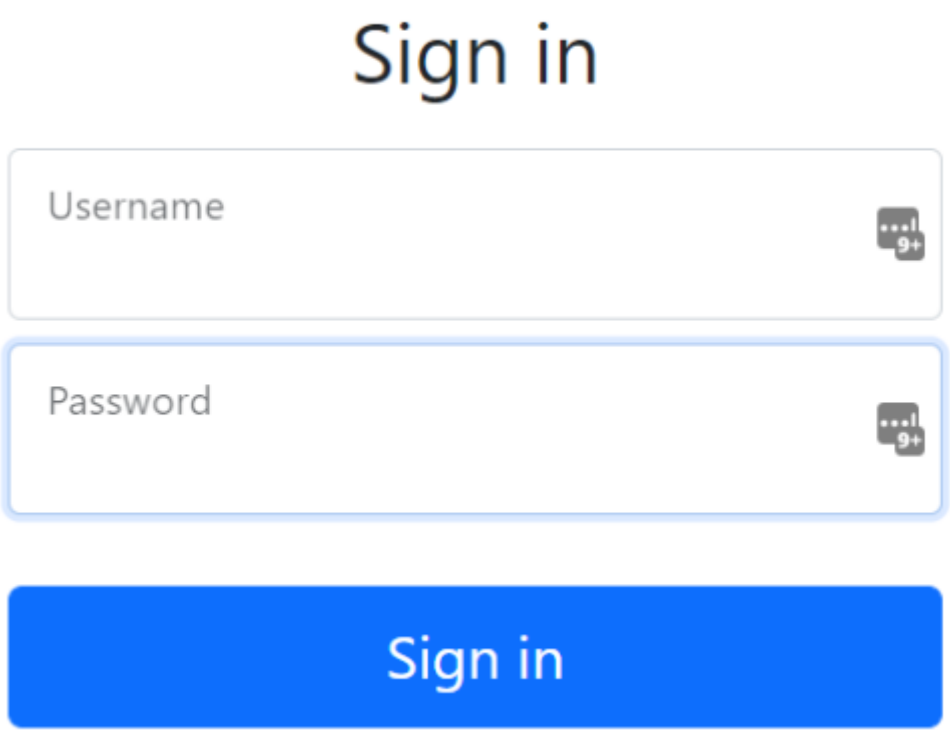
DEFINICE

Autentizace je proces ověření proklamované identity subjektu (osoby nebo zařízení).

Tento proces se uskutečňuje na základě informací poskytnutých daným subjektem, jehož identita je ověřována. Proces autentizace u privátních i veřejných počítačových systémů, jakými jsou např. počítačové sítě, často vyžaduje, aby subjekt, obvykle uživatel, použil k přihlášení pověření vydaná daným systémem. Skutečnost, že subjekt použije správné heslo, pak prokazuje pravost jeho identity. Nejběžnější metodou ověřování je kombinace uživatelského jména a hesla. Existují i další možnosti autentizace, např. pomocí biometrických údajů, čipových karet, jednorázových tokenů apod.

Ve většině případů je při ověřování nutné předložit pověření nebo aktivum, které dokládá tvrzení, že daný subjekt je tím, za koho se vydává. Aktiva nebo pověření mohou být založena na řadě odlišných charakteristik, případně jejich specifických kombinacích, které však jednoznačně prokazují identitu subjektu (např. uživatele služeb nebo původce zprávy). Pro ověření skutečné identity subjektu se používají následující základní metody:

- **podle toho, co uživatel zná** (zná správnou kombinaci uživatelského označení a hesla) – jedná se o duševní vlastnictví uživatele, např. heslo, které zná jak uživatel na jedné straně, tak ověřovatel na straně druhé. Ačkoli jde o poměrně efektivní řešení správy přístupů z pohledu finančních nákladů, je toto řešení i poměrně zranitelné, např. pokud uživatel své heslo zapomene. Trpí i dalšími nedostatky, jakými je např. nízká úroveň zabezpečení při ukládání souborů s hesly ze strany správců systému. Uživatel také může používat stejné heslo pro přístup k různým systémům. Obecně do této kategorie metod ověření spadají hesla, různé přístupové fráze a osobní identifikační čísla tzv. **PIN** (***Personal Identification Number***).

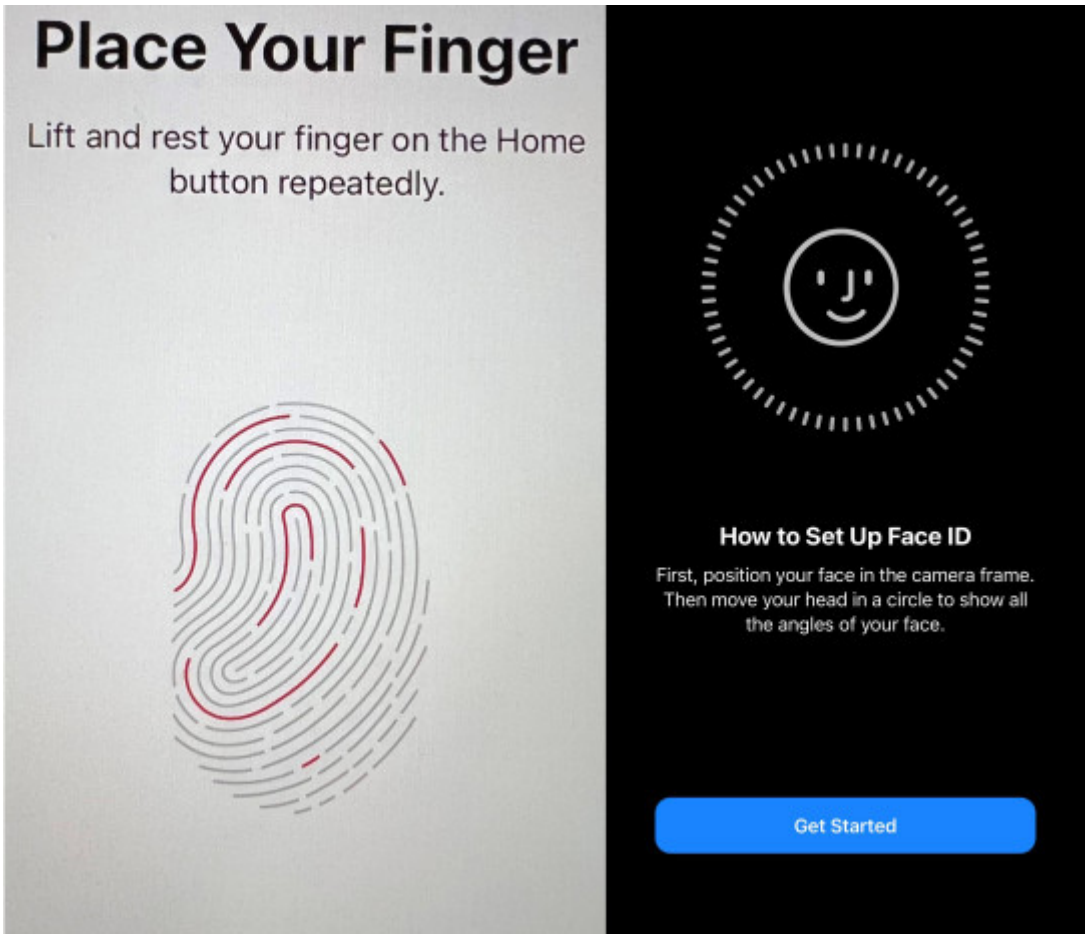


Obr. 1. Příklad přihlášení uživatele pomocí uživatelského jména a hesla.

- **podle toho, co uživatel má** (technický prostředek, který uživatel vlastní – hardwarový klíč, smart card, soukromý klíč apod.) – ve své podstatě se může jednat o jakýkoli typ vydaného nebo získaného samoidentifikačního tokenu nebo tagu, včetně čipových karet, hardwarových tokenů, mobilních telefonů a řady dalších prostředků. Vzhledem k tomu, že tyto technické identifikátory je mnohem obtížnější napodobit a následně je zneužít, je tato forma ověření uživatele bezpečnější než předchozí metoda (co uživatel zná). Např. odcizit čipovou kartu je mnohem obtížnější než jen zapamatovat si její číslo.

Obr. 2. Příklady hardwarových tokenů pro typ ověřování pomocí technického prostředku.

- **podle toho, čím uživatel je** (uživatel disponuje unikátními biometrickými identifikátory, které lze prověřit) – jedná se tedy o přirozeně získaný unikátní tělesný znak. Tento druh ověřování se většinou označuje jako **biometrie**. Ačkoli je použití biometrických údajů ve své podstatě jednoduché, problémem mohou být finanční náklady spojené s pořízením spolehlivých biometrických čteček. Biometrickými údaji mohou být např. otisky prstů, snímky oční duhovky či sítnice, vzory DNA nebo rozpoznání charakteristických rysů obličeje.



Obr. 3. Příklady biometrického ověřování pomocí chytrého telefonu.

VÝHODY

Pokud systém systematicky vyžaduje několik různých typů, resp. různých kombinací ověření identity uživatele, lze tímto způsobem dosáhnout velmi robustního zabezpečení.

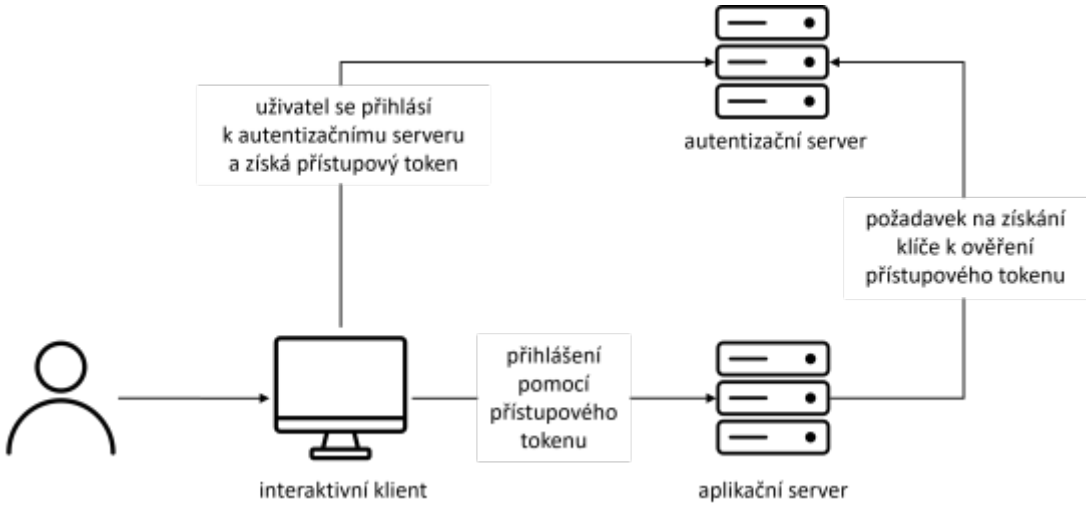
NEVÝHODY

Příliš časté, resp. robustní ověřování může mít zcela opačný účinek, a tím je diskomfort uživatele.

Interaktivní prvek

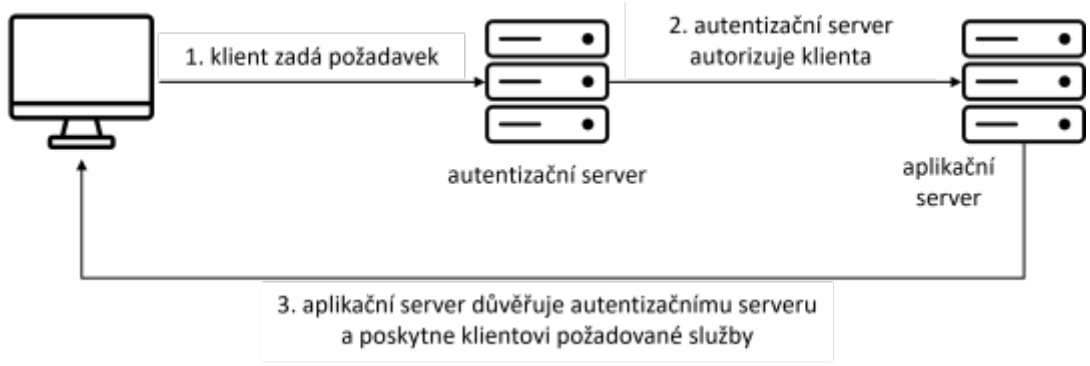
Jiným způsobem, jak rozlišit metody ověřování, je dle typu využitých prostředků v procesu identifikace uživatele. Metody lze rozdělit do tří následujících kategorií:

- **Základní ověřování na serveru.** Server například uchovává identifikační soubor s hesly, uživatelskými jmény a některými dalšími doprovodnými ověřovacími údaji. V tomto případě se jedná o nejrozšířenější metodu ověřování uživatelů. Tento způsob má však několik nedostatků, které nepočítají s tím, že např. uživatel může své heslo zapomenout nebo že mohou být hesla, resp. ověřovací údaje uloženy nekorektním způsobem.



Obr. 4. Základní ověřování na serveru

- **Metoda výzva-odpověď** je metodou ověřování, při níž server nebo jiný ověřovací systém vydá hostiteli požadujícímu ověření výzvu a čeká na jeho odpověď. Příkladem této metody je např. použití tzv. nonce – jedná se o textový řetězec, číslo nebo sekvenci bitů které nemají žádný hlubší význam, ale slouží k jednorázovému procesu ověření identity, a která zároveň svým charakterem zajišťují, že tato data jsou v procesu ověření použita právě a pouze jednou.
- **Centralizované ověřování** označuje systém, ve kterém server ověřuje, autorizuje a kontroluje uživatele sítě. Tyto tři procedury se provádějí v reakci na aktivitu serveru. Příkladem takového ověřování je např. systém Kerberos.



Obr. 5. Centralizované ověřování

Interaktivní prvek

KAPITOLA 2

Proces ověřování a jeho prvky

Identifikace vyžaduje proces ověřování (autentizace) a zahrnuje následující prvky:

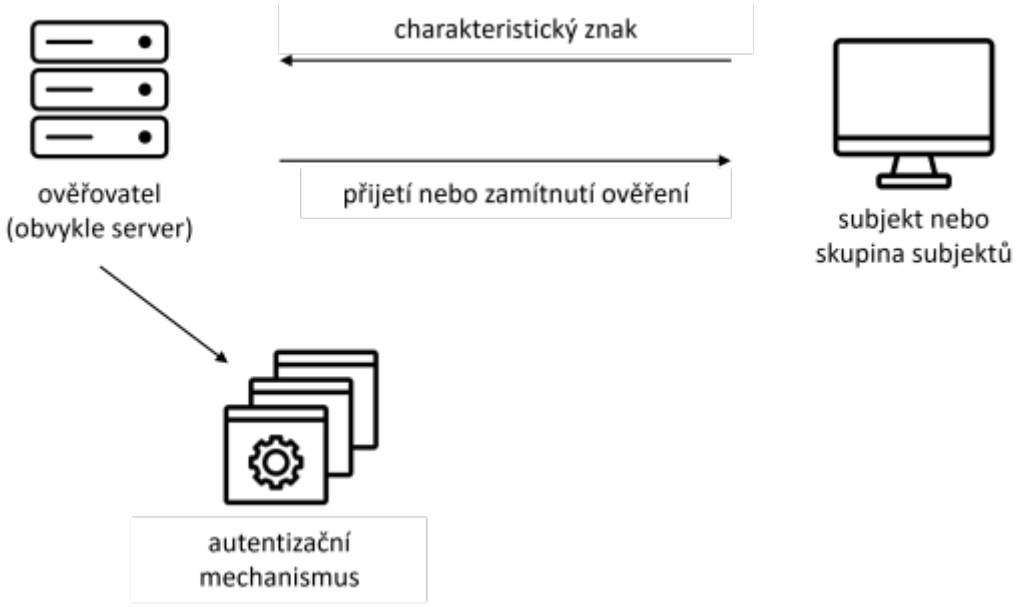
- subjekt nebo skupina subjektů, které usilují o ověření pravosti (identity);
- rozlišovací znak od subjektu nebo skupiny subjektů, jejichž pravost se jím ověřuje;
- autentizátor / ověřovatel (obvykle jím bývá server);
- ověřovací mechanismus ověřující a jednoznačně určující pravost rozlišovacího znaku;
- mechanismus řízení přístupu pro přijetí nebo odmítnutí ověření.

Prvním prvkem jsou velmi často **osoby, procesy** nebo **zařízení,** které chtějí získat přístup do systému. Pokud jednají individuálně, musí být připraveny prokázat ověřovateli, že jsou oprávněny využívat požadovaný systémový prostředek.

Druhým ověřovacím prvkem je **rozlišovací znak** uživatele. Konkrétně byly zmíněny výše a lze je dělit do kategorií **podle toho, co uživatel zná**, **podle toho, co uživatel má** a **podle toho, čím uživatel je**. Některé z těchto prvků však nemusí samostatně stačit k jednoznačnému ověření subjektu. Využitím kombinace více rozlišovacích prvků lze vylepšit samotný proces ověřování a zároveň tak poskytnout vyšší jistotu v identifikaci subjektu a vyšší míru důvěryhodnosti procesu ověřování.

Úkolem autentizátora je jednoznačně a automaticky zkontrolovat pověření entity a určit, zda má tato entita povolen přístup k požadovanému systémovému prostředku. Po odeslání požadavku na ověření si autentizátor vyžádá pověření k dokončení procesu ověřování. **Autentizátor**, poté co shromáždí všechna potřebná data, je odešle autentizačnímu mechanismu. Autentizátorem může být server určený uživatelem, virtuální privátní síť (VPN), brána/firewall, webový server, vyhrazený podnikový server, nezávislá ověřovací služba nebo jiný typ globální služby umožňující ověření identity. Bez ohledu na to, který prostředek je použit jako **autentizátor**, však musí být zajištěno úspěšně dokončení procesu autentizace, jehož výsledkem je určitá výsledná hodnota, například token, kterou lze následně použít ke zjištění informací o oprávněném uživateli.

Příklad tohoto procesu ověřování je znázorněn na Obr. 5.



Obr. 6. Základní proces ověřování a jeho prvky

Ověřovací mechanismus se skládá ze tří částí, které společně zajišťují přítomnost rozlišovacích znaků uživatele:

- vstupní zařízení,
- distribuční systém (mezi uživatelem a ověřovatelem) a
- ověřovatel.

Interaktivní prvek

Vstupní zařízení slouží k interakci uživatele s mechanismem ověřování. Příkladem je např. klávesnice počítače, čtečka karet, videokamera, telefon nebo podobné srovnatelné zařízení. Zachycené rozlišovací znaky identifikující konkrétní subjekt jsou následně distribuovány do místa, kde se zkontrolují, analyzují a následně se buď přijmou nebo odmítnou. Aby se však tyto znaky dostaly na toto místo, musí být přepraveny. V důsledku toho má distribuční část systému na starosti předávání dat mezi vstupní složkou a prvkem, který může ověřit totožnost osoby. Tyto informace se přenášejí u moderních systémů prostřednictvím sítě, kde lze jejich přenos adekvátními protokoly zabezpečit. Poslední složkou autentizačního systému je ověřovatel, který představuje mechanismus kontroly přístupu.

Interaktivní prvek

2.1 Typy ověření (autentizace)

V předchozím textu jsme uvedli tři faktory, které se používají při ověřování identity uživatele. Bylo též konstatováno, že ačkoli jsou všechny tyto faktory použitelné, některé z nich bohužel trpí určitou mírou zranitelnosti. Tabulka 1 souhrnně ukazuje nedostatky jednotlivých faktorů.

Tabulka 1. Kategorie ověření a jejich nedostatky

Faktor	Příklady	Zranitelnosti
co víme	heslo, PIN	lze zapomenout, uhodnout, duplikovat, snadno získat v případě podvodu (např. phishing)
co máme	tokeny, smart karty, jednorázové heslo zaslané na vaše telefonní číslo	mohou být ztraceny, odcizeny, duplikovány
čím jsme	otisk prstu, snímek oční duhovky nebo sítnice, rozpoznání charakteristických rysů obličeje	neodvolatelná – nelze je změnit v případě zneužití

NEVÝHODY

Bylo též uvedeno, že první dva faktory, „co víme“ a „co máme“, mohou ověřovateli působit jisté obtíže, jelikož poskytnuté informace mohou být nepřesné. Jinými slovy to znamená, že mohou být nedůvěryhodné, protože tyto faktory podléhají řadě známých problémů, včetně možnosti ztráty, padělání nebo snadnému napodobení. Údaje mohou být také zapomenuty, resp. údaje a technické prostředky mohou být sdíleny nebo odcizeny.

Interaktivní prvek

2.1.1 Vícestupňová (multifázová) autentizace

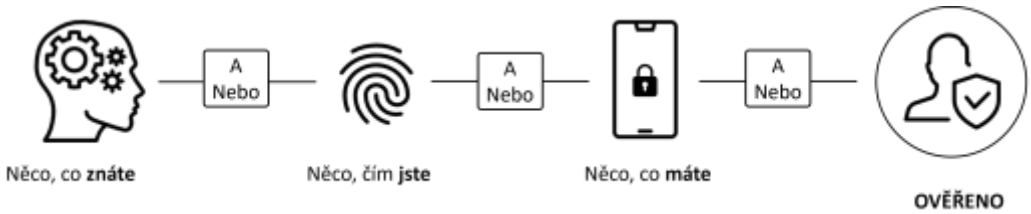
DEFINICE

Vícestupňová (multifázová) autentizace **MFA** (***Multi-Factor Authentication***) využívá kombinaci minimálně dvou různých faktorů (jednotlivé dílčí faktory jsou uvedeny výše). Dvoufázová autentizace **2FA** (***Two Factor Authentication***) je v principu stejná jako multifázová, ale využívá kombinace přesně dvou dílčích faktorů ověření.

Pokud se v dnešní době používá MFA, jedná se téměř vždy o 2FA. Prvním faktorem bývá obvykle heslo nebo PIN (tzn. faktor, co znáte), druhým pak obvykle bankovní karta (její číslo), ověřovací **SMS** (***Short Message Service***) nebo kód vygenerovaný aplikací (tzn. faktor, co máte – např. vaše mobilní zařízení). Použití otisků prstů, skenů sítnice apod. (tzn. faktor, čím jste, tj. vaše biometrické údaje) jsou další volitelnou možností, používají se však méně často, protože je zapotřebí dalšího hardwaru, což bývá

spojeno s dalším náklady.

Vícestupňová autentizace je vhodným způsobem, jak zmírnit riziko a snížit pravděpodobnost zneužití citlivých údajů. Podívejme se například na kombinaci hesla a kódu získaného prostřednictvím mobilní aplikace. Útokem na konkrétní webovou stránku, kdy získá útočník správné přístupové údaje (kombinaci uživatelského jména a hesla), případně tuto kombinaci získá jiným způsobem, přesto nebude schopen se přihlásit (i když může zadat správné uživatelské jméno a heslo), protože není schopen ještě zadat kód vygenerovaný mobilním zařízením. Odcizená kombinace jména a hesla se tak stává nepoužitelnou (v případě, pokud útočník neukradne i mobilní zařízení, ale v tomto případě se již nejedná o škálovatelný útok, a proto pro většinu lidí nepředstavuje vážné riziko). Správci systému mohou mezitím zjistit nárůst neúspěšných pokusů o přihlášení a cíleně požádat konkrétního uživatele o změnu hesla nebo všechny uživatele, pokud byl jejich systém napaden a všechna hesla unikla.



Obr. 7. Vícestupňová (multi-fázová) autentizace

KAPITOLA 3

Ověřování heslem

Technika ověřování heslem je nejběžnější a nejsnadněji použitelnou. U celé řady systémů je obvykle nastavena jako výchozí metoda autentizace. Opakovaně použitelná hesla, jednorázová hesla **OTP** (***One-Time Passwords***), hesla s výzvou k odpovědi, případně hesla s kombinovaným přístupem jsou příklady technik autentizace založených na ověřování heslem.

Opakovaně použitelná hesla

Při ověřování identity pomocí opakovaně použitelných hesel lze rozlišit dvě formy ověřování, a to buď ověřování uživatele, nebo ověřování klienta.

- **Ověřování uživatele** je nejrozšířenějším druhem ověřování a většina uživatelů jej pravděpodobně zná. Iniciátorem je vždy uživatel, který na server zašle požadavek na ověření a autorizaci pro přístup k určitému systémovému prostředku. Pokud server požadavek akceptuje, vyzve uživatele k zadání uživatelského jména a hesla. Zadané údaje server následně porovná s kopiemi údajů ve své databázi. V případě jejich shody je poskytnuta požadovaná autorizace.
- **Ověřování klienta** – uživatel obvykle žádá server o ověření své identity a následně o autorizaci pro přístup k systému nebo sadě systémových prostředků. Ověření identity uživatelů tedy automaticky nezaručuje přístup uživatelů k libovolnému systémovému prostředku. Oprávnění uživatele využívat požadované prostředky pouze ve vymezeném rozsahu musí být součástí ověření jeho identity. Tento druh autentizace se označuje jako ověřování/autentizace klienta Nejprve je zjišťována identita uživatelů a následně jim je umožněn řízený přístup k systémovým prostředkům.

Jelikož jsou tyto metody ověřování nejčastěji používané, bývají také i nejčastěji zneužívané.

NEVÝHODY

Navíc jsou tyto metody nespolehlivé, protože lidé své přístupové údaje zapomínají, zapisují si je, sdílejí je a někdy je lze snadno i uhodnout, protože jsou používána jednoduchá hesla. Jsou tedy také zranitelná vůči sledování a prolomení. Slabá hesla (např. krátká, popř. s jednoduchou strukturou) jsou o to víc zranitelná pro dnešní výkonné počítače, které je mohou prolomit pouhou hrubou výpočetní silou, tzv. *brute-force attack*.

Jednorázové ověřování heslem

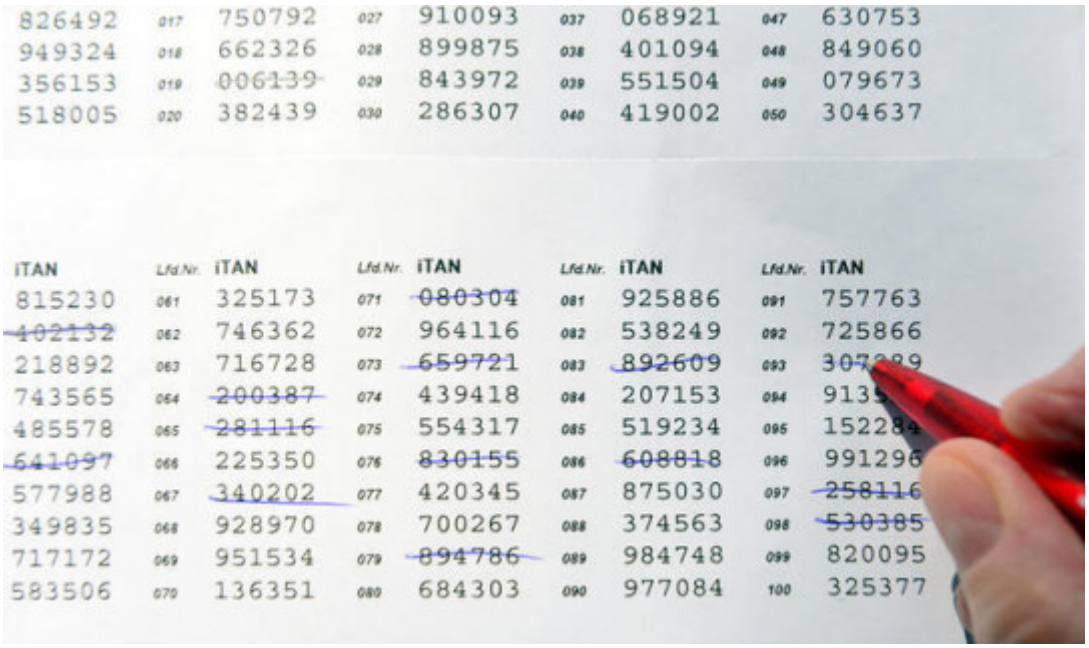
Ověřování relací je jiný název pro jednorázové ověřování heslem. Na rozdíl od opakovaně použitelných hesel, která lze použít vícekrát, se jednorázová hesla použijí pouze, a právě jen jednou a poté se zahodí.

VÝHODY

Jednorázová hesla jsou generována náhodně pomocí spolehlivých generátorů náhodných čísel. Tím se snižuje pravděpodobnost jejich uhodnutí. V mnoha případech jsou ještě před odesláním zašifrována tak, aby se omezila možnost jejich zneužití.

Jednorázová hesla mohou mít různou podobu. Příkladem jsou hesla typu **SIKey** a tzv. tokeny. **SIKey** je systém vytvářející jednorázová hesla definovaný v dokumentu RFC 1760.

Dalším příkladem je identifikátor **TAN** (***Transaction Authentication Number***), který se v minulosti využíval v Německu. (viz Obr. 8). Jednorázová hesla jsou sice obvykle bezpečnější, ale mají i řadu nevýhod, včetně problémů se synchronizací způsobených rozdílem mezi časovým údajem v hesle a systémovými hodinami. Heslo nelze použít, pokud jsou tyto dva časové údaje rozdílné, tj. mimo fázi.



Obr. 8. Identifikátory TAN jsou příkladem jednorázových hesel OTP

Některé typy jednorázových hesel (např. SMS kódy nebo kódy generované aplikacemi) se obvykle používají jako druhý faktor ve dvoufázové autentizaci typu 2FA.

Hesla typu výzva-odpověď

DEFINICE

Autentizační metodou založenou na ověřování hesla je i tzv. heslo s výzvou k odpovědi, při níž ověřovatel vyzve uživatele, který žádá o ověření, aby se přihlásil. Aby byl uživatel ověřen, musí uvést správnou odpověď na výzvu/dotaz systému.

Formát výzvy závisí na její implementaci v systému a může se tedy systém od systému lišit. Může to být požadavek na zadání hesla, čísla (např. PIN), provedení určitého výběru z nabídky nebo nonce. Osoba, která chce být ověřena, musí bezpodmínečně odpovědět na výzvu systému. V současné době se odpovědi

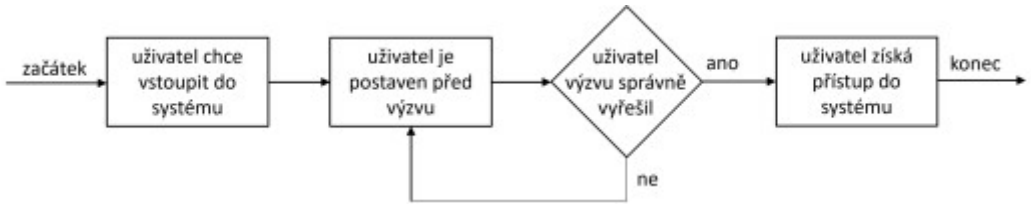
odesílají prostřednictvím jednosměrné služby a tokenu pro heslo označovaného jako asynchronní token. Jakmile server obdrží odpověď uživatele, heslo následně dvakrát překontroluje.

Nejčastěji se ověřování typu výzva-odpověď používá u distribuovaných systémů. Přestože je tento typ ověřování poměrně rozšířený, trpí určitými nedostatky, např. vyžaduje aktivitu ze strany uživatele a útočník jej může napadnout metodou pokus-omyl (*trial-and-error attack*). Aktivitou uživatele je chápána jeho schopnost najít požadovaný prvek na obrazovce, což záměrně nebývá jednoduché. Odpověď poté musí uživatel zadat v omezeném (obvykle krátkém) čase.

V závislosti na požadované úrovni zabezpečení může být od uživatele požadováno, aby si zapamatoval delší odpověď, může být vyžadován její přepis a následný opětovný zápis. Tento způsob však velmi často vede k chybám.

ZAJÍMAVOST

Někteří výrobci se pokusili snížit zátěž uživatele spojenou s procesem zapamatování a opětovného zadávání dlouhých řetězců tím, že většinu nezbytných kroků automatizovali, a to buď možností vyjmout a vložit (*Cut & Paste*) výzvu a odpověď, nebo nízko úrovňovým automatizovaným procesem, který omezuje reakci uživatele výhradně na odpovědi typu ano/ne.



Obr. 9. Proces ověřování u systému výzva-odpověď

Za zmínku stojí také to, že hesla typu výzva-odpověď v jejich nejjednodušší (základní) podobě lze velmi snadno zneužít, protože správná hesla lze poměrně snadno odhalit. Hesla mohou také být zachycena, pokud jsou přenášena v otevřeném nešifrovaném formátu. Pokud však není heslo přenášeno v otevřeném formátu (tzn. je zašifrované), nepředstavuje to výrazné bezpečnostní riziko.

Interaktivní prvek

3.1 Problémy se zabezpečením hesel

Úniky dat jsou v případě kybernetických útoků jedním z nejčastějších cílů, na které útočníci svou aktivitu cíleně zaměřují. Proto musí být autentifikační mechanismy založené na heslech stále vylepšovány.

Jedinečnost hesla je jednou z jeho nejdůležitějších vlastností. Mnohá hesla však tuto klíčovou podmínku zdaleka nesplňují. Nejoblíbenější hesla a fráze používané lidmi po celém světě jsou uvedené v tabulce 2.

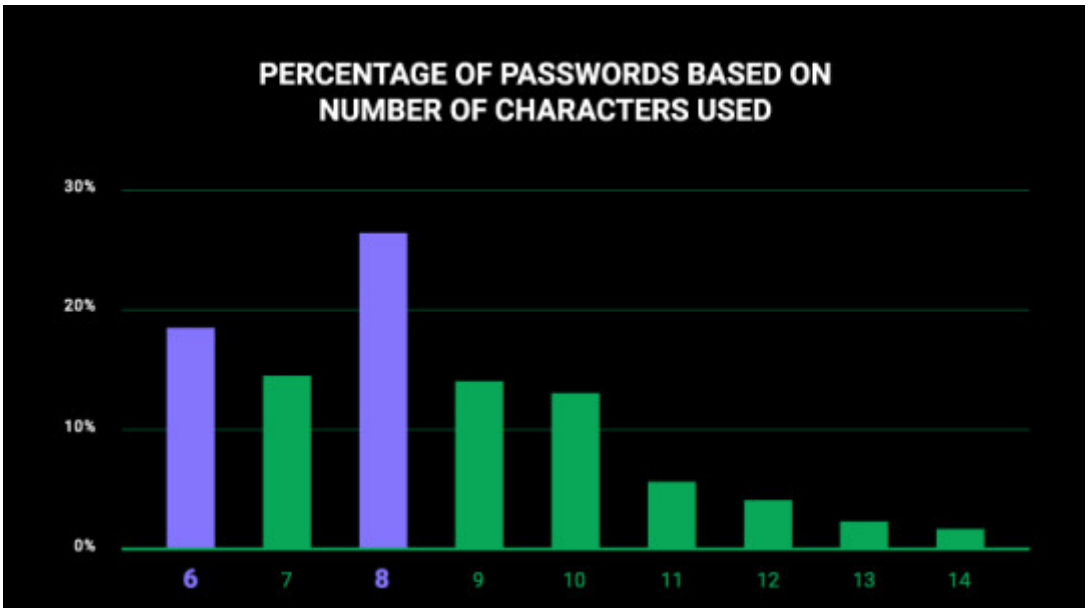
Tabulka 2. Deset nejčastěji používaných hesel, zdroj: cybernews.com

Heslo
123456
123456789
qwerty
password
12345
qwerty123
1q2w3e
12345678
111111
1234567890

NEVÝHODY

Kromě toho existují i jiné problémy s hesly. Mnoho lidí se snaží přístup na webové stránky navázat na něco, co si snadno zapamatují, a vytváří tak jednoduché a snadno zapamatovatelné kombinace. Nicméně to však neznamená, že je takové heslo jedinečné, opak je spíš pravdou.

Investigativní skupina Cybernews prozkoumala přibližně 15 miliard záznamů a rozdělila je do několika kategorií a skupin. Z výsledků jejich analýzy následně plyne, že problematickými jsou určité vlastnosti hesel, pokud obsahují údaje týkající se přímo uživatele. Dále zkoumali délku hesel, resp. počet použitých znaků. Většina používaných hesel bohužel měla délku 8 nebo méně znaků.



Obr. 10. Statistiky délky hesel, zdroj: cybernews.com

Existují však účinné metody, které umožňují vytvořit opravdu silné heslo. Např. použitím slova „*heat*“ jako prvku hesla může být jednoduchým heslem heslo „*letsgoheat*“ (10 znaků), zatímco složitějším heslem může být „*heatromearsenalhjamesp*“ (22znakové slovní spojení). Lidé často vytvářejí bezpečná hesla pomocí mnemotechnických pomůcek, protože to má několik nesporných výhod. Tato hesla jsou pak často relativně dlouhá a obsahují náhodná slova bez logického vztahu mezi nimi, takže si je člověk snadněji zapamatuje, ale pro algoritmus je obtížnější je prolomit.

3.2 Útoky na hesla

Hesla lze napadnout různými způsoby. Obecně je lze klasifikovat následujícím způsobem:

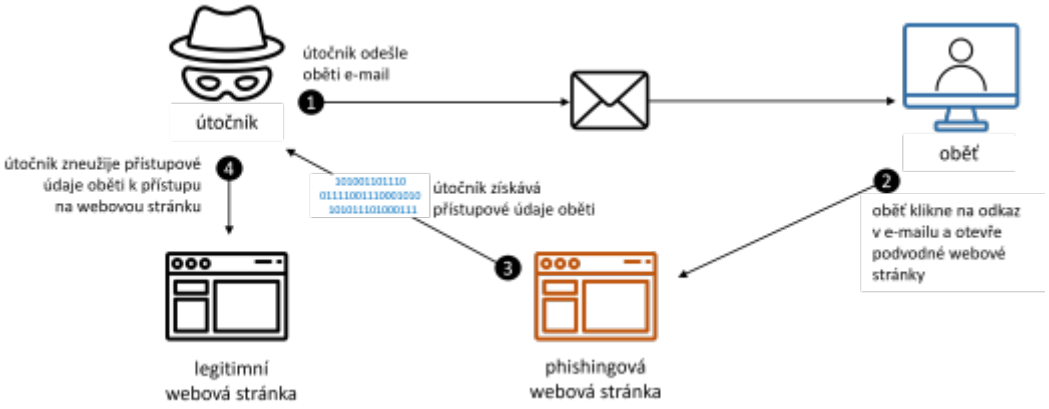
- Neelektronické útoky nevyžadují k prolomení hesla hluboké technické dovednosti. Příkladem takových útoků je např. skryté sledování potenciální oběti (tzv. *Shoulder Surfing*), sociální inženýrství a případně analýza dat z odpadků oběti (tzv *Dumpster Diving*).
- Útoky elektronického typu již vyžadují jisté technické dovednosti. Příkladem takových útoků jsou slovníkové útoky (*Dictionary Attacks*), útoky hrubou silou (*Brute-Force Attacks*) a útoky pomocí duhových tabulek (*Rainbow Table Attacks*).

3.2.1 Neelektronické útoky

DEFINICE

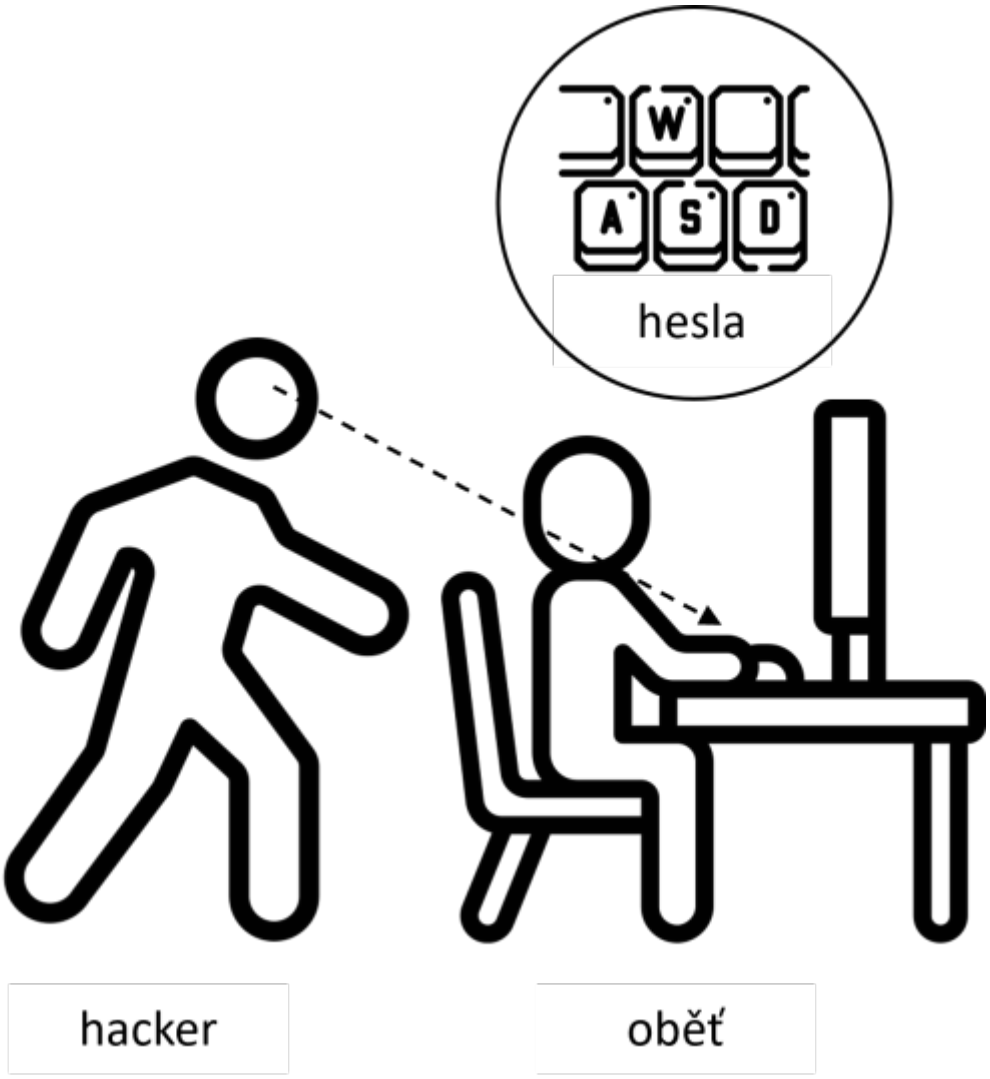
Sociální inženýrství je typ útoku, při kterém se útočník snaží zneužít přirozené důvěry lidí vůči ostatním. Zneužitím této důvěry útočník často velmi rychle získá citlivé osobní nebo přímo přihlašovací údaje oběti, které následně využije pro přístup k účtu oběti.

Útoky typu *phishing*, *pharming* a *whaling* jsou jen některými vybranými příklady. Upozorňujeme, že některé z těchto útoků vyžadují jisté technické dovednosti (např. *phishing*).



Obr. 11. Proces phishingového útoku

Při útoku založeném na sledování oběti stojí útočník za vámi a pozoruje, jaké zadáváte přihlašovací údaje, které následně zneužije pro přístup k vašemu účtu.



Obr. 12. Příklad útoku založený na sledovaní oběti

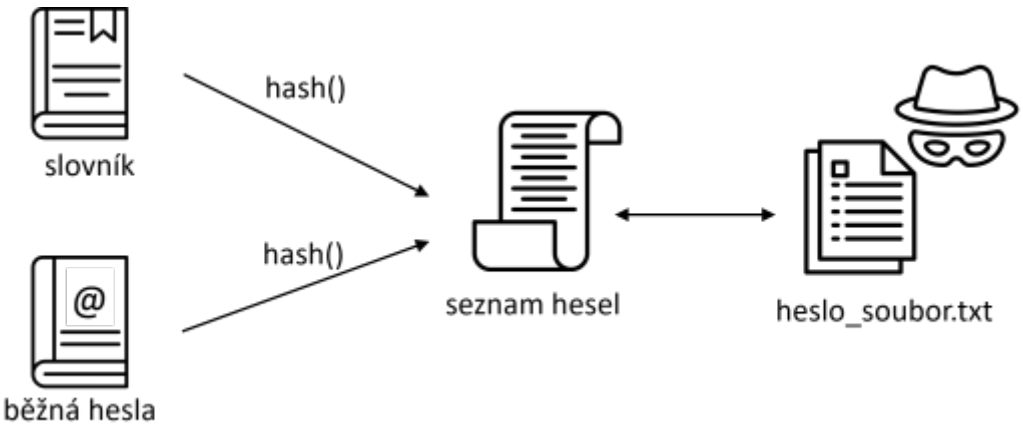
Při útoku založeném na analýze dat z odpadků oběti se útočník snaží objevit něco cenného, jako je např. heslo nebo PIN k vaší kreditní kartě.

3.2.2 Elektronické útoky

DEFINICE

Slovníkový útok je útok, při kterém se útočník pokouší dostat do systému chráněného heslem pomocí vhodně zvoleného slova ze slovníku s cílem uhodnout heslo pro daný systém.

Tento proces zahrnuje testování všech vytipovaných řetězců z předem připraveného seznamu. Historicky se při těchto útocích používala slovníková slova (odtud výraz slovníkový útok).



Obr. 13. Jak funguje slovníkový útok

DEFINICE

Slovníkový útok zkouší takové výrazy, které se z hlediska úspěchu jeví jako nejpravděpodobnější.

Řada lidí má tendenci volit příliš krátká hesla. Často jde o obyčejná slova nebo běžná hesla, případně varianty získané například přidáním číslice nebo interpunkčního znaménka. A právě proto bývají slovníkové útoky tak často úspěšné.

Vzhledem k tomu, že běžně dostupné slovníky pokrývají většinu typických strategií pro vytváření hesel, je obtížné čelit slovníkovým útokům pomocí nástrojů, které generují vzory umožňující prolomení hesla. Bezpečnějším způsobem je použití nástroje pro správu hesel nebo manuální metody vytvoření delšího hesla (15 a více písmen) nebo víceslovného hesla realizovaného náhodně.

Útoky hrubou silou

DEFINICE

Zjednodušeně řečeno, **útok hrubou silou** je metoda prolamování hesel, při které útočník zkouší co nejvíce možných kombinací hesel pomocí vytvořené sady parametrů.

Webová stránka může například stanovit omezení, dle kterého musí být heslo dlouhé 8 až 16 znaků. V základní variantě může proces prolomení hesla začínat číslem „00000000“. Pak může vyzkoušet „00000001, 00000010, 00000100“ a tak dále, dokud nevyčerpá všechny možné kombinace znaků.

Hesla tedy mohou obsahovat znaky:

- malá (anglická) abeceda (26 možností),
- velká (anglická) abeceda (26 možností),
- číslice (10 možností od 0 do 9),
- interpunkční znaménka nebo další speciální znaky (33 možností).

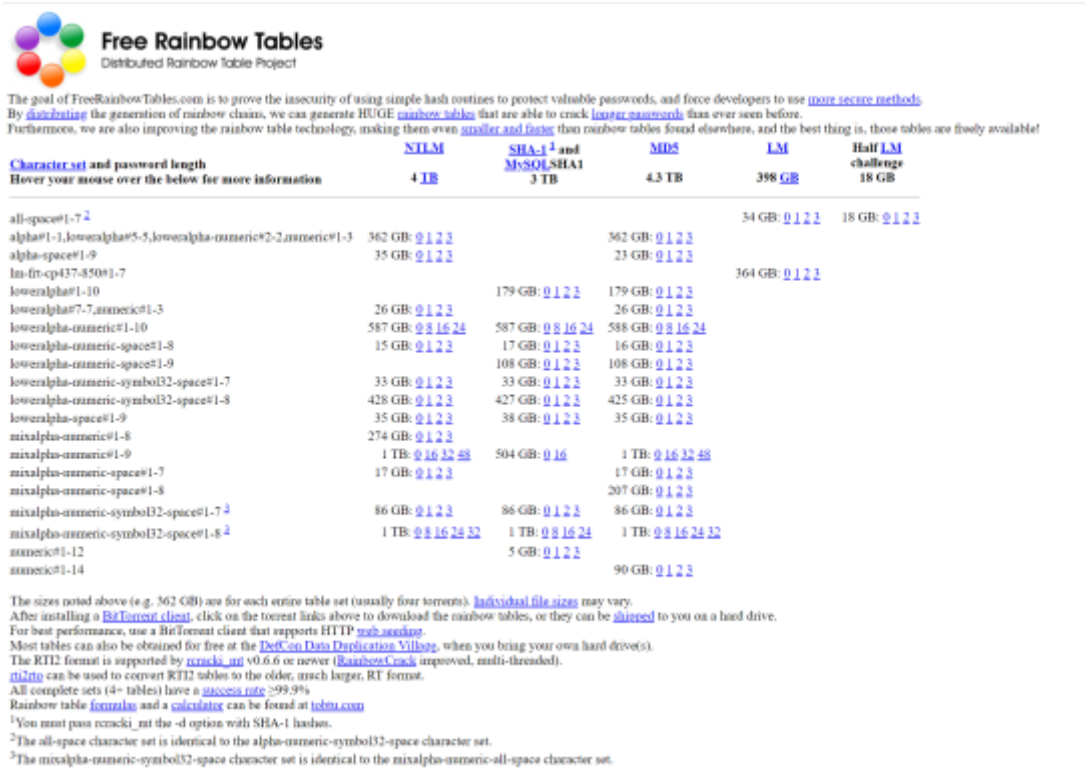
Vzhledem k výše zmíněnému pak lze vypočítat celkový počet možných hesel pro osmimístné heslo: 3 025 989 069 143 040, tedy přibližně 3 kvadriliony, přičemž každé z nich představuje samostatný pokus.

Možná si teď pomyslíte, že někdo vytvoří program, který přejde na webovou stránku, zadá vaše uživatelské jméno a heslo, stiskne tlačítko pro přihlášení a pokusí se uhádnout vaše heslo. Potom stejný proces zopakuje ještě tři kvadriliónkrát. To však není tento případ. Pokud trvá načtení stránky 2 sekundy, znamená to 2 sekundy čekání na každý další pokus o zobrazení stránky s chybným heslem. Jinými slovy, pokud webová stránka sama nezablokuje proces přihlášení po určitém počtu podezřelých pokusů, může to trvat až 9 kvadrilionů sekund, tedy 287,9 milionu let, než by útočník tímto způsobem vaše heslo prolomil. Ve skutečnosti se takový útok provádí pomocí uniklých uživatelských jmen a hesel. Tyto údaje unikly v důsledku krádeže dat (což se stává častěji, než si myslíte). Heslo je pak možné sdělit jedním ze dvou následujících způsobů:

- Vaše heslo není šifrováno a je uloženo jako prostý text v extrémně málo chráněném prostředí. Útočníkovi pak stačí heslo pouze zkopírovat a vložit (*Copy & Paste*). Pokud je vaším heslem například „heslo1“, každý, kdo je schopen přečíst obsah uniklých údajů, uvidí „heslo1“. V tomto případě je tedy útok hrubou silou zcela zbytečný, jelikož webová stránka již odevzdala vaše přístupové údaje útočníkovi doslova na stříbrném podnose.
- Vaše heslo je šifrované a není uloženo jako prostý text v rámci zabezpečeného prostředí. Pokud by webová stránka šifrovala heslo pomocí hashovací funkce SHA-256, zobrazilo by se například „heslo1“ jako 0b14d501a594442a01c6859541bcb3e8164d183d32937b851835442f69d5c94e.

Obr. 14. Útok hrubou silou na hesla

Duhové tabulky jsou speciálním typem útoku hrubou silou umožňující prolomit hesla. Je určen k prolomení hesel uložených v hash formátu. Ve své podstatě jsou duhové tabulky předem vypočítaným seznamem hashů slovníkových výrazů nebo již dříve prolomených hesel. Jsou uloženy v databázi využívající daný hash jako klíč. Existuje zde však jistý kompromis, resp. omezení mezi časem nutným k prolomení hesla a požadovanou velikostí operační paměti. Generování duhové tabulky může trvat dlouho, ale stačí jej provést pouze jednou. Po dokončení výpočtu duhové tabulky můžete vyhledat hash hesla a následně velmi rychle získat příslušné heslo. Abychom si udělali představu o obrovské velikosti těchto databází, některé duhové tabulky mohou mít velikost 7-9 TB.



Free Rainbow Tables
Distributed Rainbow Table Project

The goal of FreeRainbowTables.com is to prove the insecurity of using simple hash routines to protect valuable passwords, and force developers to use more secure methods.
By distributing the generation of rainbow chains, we can generate HUGE rainbow tables that are able to crack longer passwords than ever seen before.
Furthermore, we are also improving the rainbow table technology, making them even smaller and faster than rainbow tables found elsewhere, and the best thing is, those tables are freely available!

Character set and password length Hover your mouse over the below for more information	NTLM 4 TB	SHA-1 [1] and MySQLSHA1 3 TB	MD5 4.3 TB	LM 398 GB	Half LM challenge 18 GB
all-space#1-7 [2]				34 GB: 0 1 2 3	18 GB: 0 1 2 3
alpha#1-1,loweralpha#5-5,loweralpha-numeric#2-2,numeric#1-3	362 GB: 0 1 2 3		362 GB: 0 1 2 3		
alpha-space#1-9	35 GB: 0 1 2 3		23 GB: 0 1 2 3		
lm-frt-cp437-850#1-7				364 GB: 0 1 2 3	
loweralpha#1-10		179 GB: 0 1 2 3	179 GB: 0 1 2 3		
loweralpha#7-7,numeric#1-3	26 GB: 0 1 2 3		26 GB: 0 1 2 3		
loweralpha-numeric#1-10	587 GB: 0 8 16 24	587 GB: 0 8 16 24	588 GB: 0 8 16 24		
loweralpha-numeric-space#1-8	15 GB: 0 1 2 3	17 GB: 0 1 2 3	16 GB: 0 1 2 3		
loweralpha-numeric-space#1-9		108 GB: 0 1 2 3	108 GB: 0 1 2 3		
loweralpha-numeric-symbol32-space#1-7	33 GB: 0 1 2 3	33 GB: 0 1 2 3	33 GB: 0 1 2 3		
loweralpha-numeric-symbol32-space#1-8	428 GB: 0 1 2 3	427 GB: 0 1 2 3	425 GB: 0 1 2 3		
loweralpha-space#1-9	35 GB: 0 1 2 3	38 GB: 0 1 2 3	35 GB: 0 1 2 3		
mixalpha-numeric#1-8	274 GB: 0 1 2 3				
mixalpha-numeric#1-9	1 TB: 0 16 32 48	504 GB: 0 16	1 TB: 0 16 32 48		
mixalpha-numeric-space#1-7	17 GB: 0 1 2 3		17 GB: 0 1 2 3		
mixalpha-numeric-space#1-8			207 GB: 0 1 2 3		
mixalpha-numeric-symbol32-space#1-7 [3]	86 GB: 0 1 2 3	86 GB: 0 1 2 3	86 GB: 0 1 2 3		
mixalpha-numeric-symbol32-space#1-8 [3]	1 TB: 0 8 16 24 32	1 TB: 0 8 16 24	1 TB: 0 8 16 24 32		
numeric#1-12		5 GB: 0 1 2 3			
numeric#1-14			90 GB: 0 1 2 3		

The sizes noted above (e.g. 362 GB) are for each entire table set (usually four torrents). Individual file sizes may vary.
After installing a BitTorrent client, click on the torrent links above to download the rainbow tables, or they can be shipped to you on a hard drive.
For best performance, use a BitTorrent client that supports HTTP web seeding.
Most tables can also be obtained for free at the DefCon Data Duplication Village, when you bring your own hard drive(s).
The RTI2 format is supported by rcracki_mt v0.6.6 or newer (RainbowCrack improved, multi-threaded).
rti2rto can be used to convert RTI2 tables to the older, much larger, RT format.
All complete sets (4+ tables) have a success rate ≥99.9%
Rainbow table formulas and a calculator can be found at tobtu.com
[1]You must pass rcracki_mt the -d option with SHA-1 hashes.
[2]The all-space character set is identical to the alpha-numeric-symbol32-space character set.
[3]The mixalpha-numeric-symbol32-space character set is identical to the mixalpha-numeric-all-space character set.

Obr. 15. Velikosti duhových tabulek z freerainbowtables.com

Interaktivní prvek

Interaktivní prvek

3.2.3 Nástroje pro odhalování hesel

Nástroje na odhalování hesel jsou v dnešní době stále populárnější, proto si některé z nich projdeme. Nástroje na odhalování hesel se velmi často používají k testování síly hesla nebo přímo ke spuštění nepřátelského útoku. Existuje řada online a off-line nástrojů určených výhradně pro účely odhalování hesel. Cílem online útoků často bývají vzdálená přihlašovací rozhraní, například služby typu **SSH** (***Secure Shell***) a **RDP** (***Remote Desktop Protocol***). Na opačné straně off-line útoky se objevují po úniku souborů. Poté, co jsou hesla zveřejněna, jsou obratem využita k útoku.

Mezi dostupné nástroje patří například Hashcat, John the Ripper nebo THC Hydra.

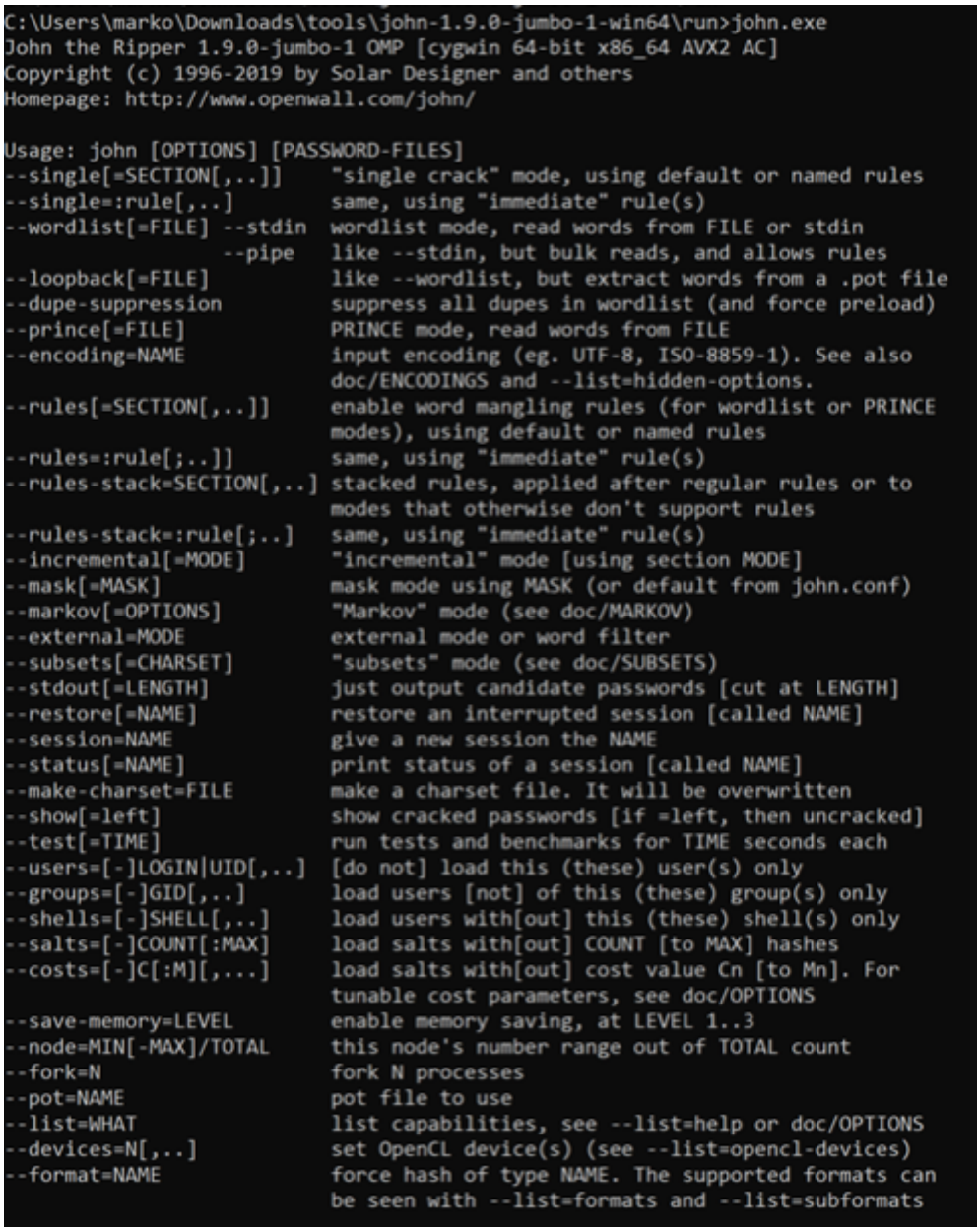
Hashcat je multiplatformní program pro obnovu hesel, který pracuje s **GPU** (***Graphics Processing Unit***) i **CPU** (***Central Processing Unit***). Hashcat byl vytvořen v roce 2009 (distribuován pod licencí **MIT** (***Massachusetts Institute of Technology***)) a je uznáván díky podpoře široké škály hashovacích algoritmů, jako jsou LM Hash, NT Hash, **MD4** (***Message Digest 4***), **MD5** (***Message Digest 5***) a mnoho dalších. V době svého vzniku tento program podporoval čtyři různé druhy útoků:

- Slovníkové útoky: více než 14 milionů hesel, počínaje těmi nejoblíbenějšími a konče těmi nejméně obvyklými. Uhodne heslo, vytvoří jeho hash a porovná hash s heslem, které se snaží prolomit.
- Kombinované útoky: podobné slovníkovým útokům, ale namísto použití dvojslovných seznamů jako slovníků vytváří nový seznam slov, kde každé slovo je spojené s každým jiným slovem.
- Maskovací útoky: pokud například víte, že heslo vašeho účtu je dlouhé 9 znaků a končí číslicí, pak víte, že k uhodnutí hesla bude potřeba 52*109 kombinací, což bude trvat přibližně 4 roky. Pokud však víte, že heslo začíná velkým písmenem a končí číslicí, doba se zkrátí na polovinu.
- Útoky založené na pravidlech: Hashcat dokáže určit, jaký druh hesla má vyzkoušet na základě způsobu, jakým oběť své heslo vytváří.
- Útoky hrubou silou: Hashcat vyzkouší všechny možnosti, dokud něco nenajde (což obvykle trvá dlouho, protože zkouší všechny možné kombinace).

John the Ripper vydaný pod obecnou veřejnou licencí GNU **GPL** (***General Public License***) v roce 1996 je off-line open-source nástroj na zabezpečení, audit a obnovu hesel, který podporuje stovky typů hashů a šifer. Je k dispozici pro různé platformy, což jej umožňuje použít na libovolné z nich. Jak již bylo uvedeno, tento nástroj podporuje řadu typů hash.

Spuštěním programu na různých platformách se mohou jednotlivé typy hashů lišit. Tento nástroj podporuje mnoho režimů luštění (hesel), včetně:

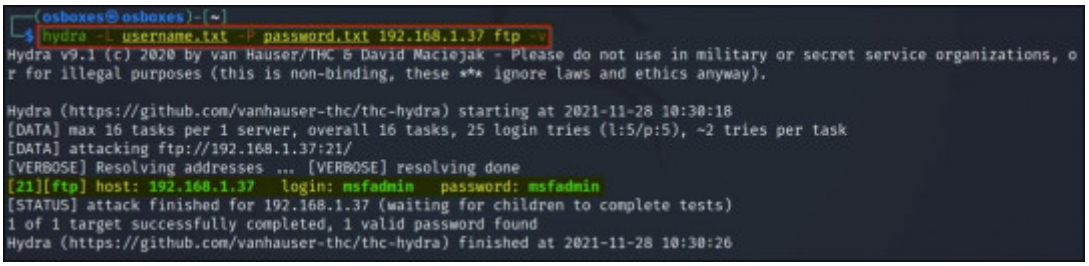
- Režim seznamu slov (slovníkový útok): V tomto režime zadáte „textový soubor“ so seznamem slov, který by měl být v ideálním případě setříděný a následně se vybraná slova porovnávají s heslem, které se pokoušíte prolomit. Je možné aplikovat různá pravidla.
- Single Crack: Jedná se o první způsob, kterým se začínají luštit hesla. Úspěšné nalezené heslo je porovnáno se všemi načtenými hesly, aby se ověřilo, zda některý z uživatelů nepoužívá stejné heslo, což celý proces značně urychluje.
- Inkrementální režim: nejvýkonnější režim luštění, který vyzkouší všechny možné kombinace, ale vzhledem k velkému počtu možných kombinací je časově velmi náročný.
- Externí režim: jedná se o funkce napsané v jazyce C, které jsou vytvořené nástrojem při jeho spuštění a výsledný kód se použije na vygenerování vhodných kandidátů na hesla.



Obr. 16. John the Ripper v akci

Program **THC hydra**, který v roce 2001 navrhl Van Hauser, je online crackovací program, který ukazuje, jak jednoduché je získat neoprávněný přístup ke vzdálenému počítači. Tento nástroj podporuje celou řadu různých protokolů včetně **FTP** (***File Transfer Protocol***), **HTTP** (***HyperText Transfer Protocol***), **HTTPS** (***HyperText Transfer Protocol Secure***), dále **MySQL** (***My Structured Query Language***), Postgress, atd. a různé platformy včetně UNIX, MacOS, Windows a mobilních zařízení. Program dokáže provést paralelní slovníkový útok, útok hrubou silou nebo hybridní útok, paralelní útok na mnoho serverů a mnoho dalších. THC Hydra je uznávána jako rychlá a účinná, avšak rychlost a účinnost je závislá na protokolu.

Hlavní rozdíl mezi programem THC Hydra a programem John the Ripper spočívá v tom, že THC Hydra je online nástroj na luštění hesel, zatímco John the Ripper je off-line nástroj.



Obr. 17. THC Hydra v akci

Interaktivní prvek

KAPITOLA 4

Různé aspekty zabezpečení hesel

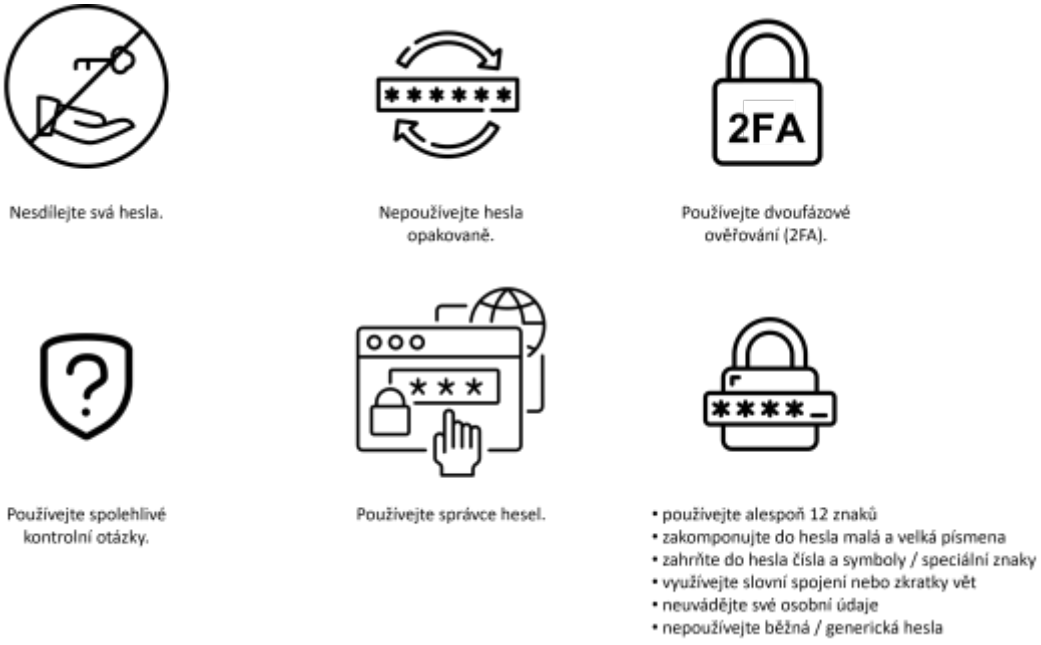
V této kapitole se budeme zabývat různými aspekty zabezpečení hesel, které lze rozdělit na aspekty:

- spojené s uživatelem,
- spojené se serverem.

Tyto aspekty zahrnují zásady a doporučení pro bezpečná hesla, dvoufázovou autentifikaci (ověření), správné a bezpečné ukládání hesel na straně serveru atd.

4.1 Pravidla a doporučené zásady pro bezpečné používání hesel

Hesla jsou stále převládající metodou autentifikace (ověření identity), jelikož jsou pro vývojáře nejjednodušší na implementaci a pro uživatele nejjednodušší na pochopení a používání. Využívání hesel však přináší i některé koncepční nedostatky (např. hesla jsou špatně zvolená, jsou snadno uhodnutelná atd.). Americký národní institut pro standardy a technologie **NIST** (***National Institute of Standards and Technology***) pravidelně aktualizuje svá doporučení pro tvorbu a správu hesel. V jednom z nedávných návrhů na změnu koncepce zabezpečení hesel navrhl, aby se uživatelé zaměřili na délku hesla a nikoli na jeho složitost (kombinace speciálních znaků, čísel, malých a velkých písmen), protože složitá hesla se špatně pamatují. Uživatelé totiž mají zpravidla tendenci dosahovat složitosti předvídatelnými způsoby (např. přidáním číslice 1 na konec hesla). Jedním ze způsobů, jak dosáhnout požadované délky, jsou i na první pohled nesmyslné heslové fráze, v nichž jsou slova v posloupnosti, která nedává žádný smysl. Ze stejného důvodu již NIST nedoporučuje při tvorbě hesla zavádět striktní pravidla pro vyžadované kombinace znaků. Zcela zásadně však doporučují pravidelně porovnávat hesla (nebo alespoň všechna nová hesla) se seznamem kompromitovaných hesel, aby bylo možné identifikovat již odhalená nebo slabá hesla. Doporučená minimální délka hesla je 12 znaků. Zatímco v minulosti se doporučovalo hesla pravidelně měnit, dnes se to již nedoporučuje, protože je méně pravděpodobné, že si uživatelé budou svá hesla po změnách pamatovat a namísto toho začnou používat stejná hesla s jen drobnými úpravami.



Obr. 18. Doporučené zásady pro používání hesel

Pro bezpečné vlastnoručně vytvořené heslo je nutné vzít v úvahu následující požadavky:

- použijte alespoň 12 znaků,
- využijte malá i velká písmena,
- zahrňte do hesla čísla a symboly / speciální znaky,

- využívejte slovní spojení nebo zkratky vět,
- neuvádějte své osobní údaje,
- nepoužívejte běžná / generická hesla.

Pamatujte si, že heslo NESMÍ obsahovat osobní údaje, jako je datum narození, jméno domácího mazlíčka, vaše jméno nebo e-mailovou adresu.

Přestože můžete pomocí různých technických opatření zajistit, aby si uživatelé volili silná hesla, není však možné kontrolovat, co uživatelé s těmito hesly dělají. Mohou si je totiž napsat na papír vedle počítače, sdílet je s dalšími lidmi nebo je používat i pro jiné účty. Zvláště nebezpečná je především poslední možnost, protože používání stejného hesla pro více účtů ohrozí všechny účty najednou, pokud dojde k prolomení některé ze služeb, které tato identická hesla využívají. To znamená, pokud používáte stejné heslo pro přístup do knihovny a zároveň třeba k e-mailovému účtu a někdo prolomí zabezpečení knihovny (což by mělo být mnohem snazší než na serverech velkého poskytovatele e-mailových služeb - např. Google) a heslo ukradne, může tyto informace zneužít pro přístup k vašemu e-mailovému účtu. Jediným skutečným řešením, jak takovým nevhodným praktikám zabránit, je pravidelná osvěta uživatelů. Uživatelé by proto měli být poučeni o tom, jak vytvářet silná hesla, aby si je nezapisovali na veřejně přístupná místa a nikdy nepoužívali stejná hesla u různých systémů či služeb.

Existují i další doporučené zásady zvyšující ochranu hesel na straně uživatele:

- používejte odlišná hesla pro různé aplikace (webové stránky)
- používejte slovní spojení (věty)
- používejte správce hesel
- používejte dvoufázové ověření identity 2FA

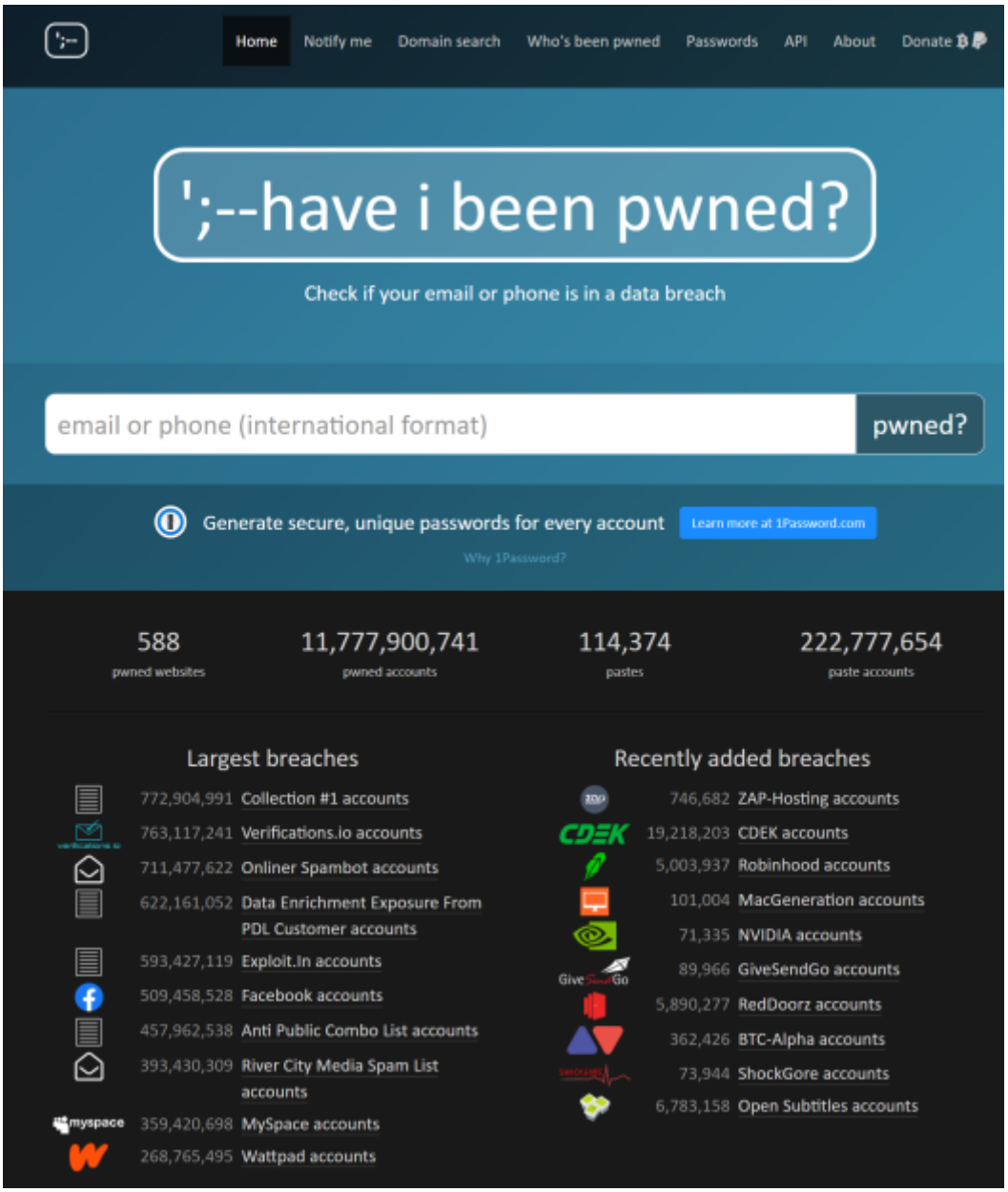
Ačkoli se to může zdát nepodstatné, používání stejného hesla na různých webových stránkách je riskantní. Úniky osobních údajů ze spotřebitelských webových stránek jsou stále častější. Pokud jsou vaše údaje odcizeny ze sociálních sítí a vy používáte identické heslo i v aplikaci internetového bankovnictví a na stránkách pro online nakupování, útočník tím získá volný přístup na všechny tyto stránky. Existují aplikace a programy, které vás mohou upozornit na to, že vaše hesla byla součástí úniku dat. Pokud došlo k úniku vašich informací, může vás na to upozornit například aplikace *Password Manager* od společnosti Google.

Interaktivní prvek

Dalším osvědčeným pravidlem je používat v hesle více než jedno slovo. **Přístupová fráze** je sled slov, který na první pohled vypadá jako věta, ale který by neměl mít žádný hlubší význam. Vaše přístupová fráze by neměla obsahovat snadno dostupné osobní údaje, podobně jako je tomu u bezpečných hesel. Pro vytvoření náhodného řetězce slov pro uživatele lze rovněž využít generátory.

Interaktivní prvek

V neposlední řadě můžete také využít různé online služby a zkontrolovat si tak, zda bylo vaše heslo prolomeno. Jednou z nejznámějších služeb tohoto typu je webová stránka http://haveibeenpwned.com. Existují však i další podobné internetové služby, jako je výše uvedená.



Obr. 19. Webové služby typu „';--have I been pwned?“

4.2 Správci hesel

Kromě dvoufázového ověřování je vhodným způsobem zabezpečení hesel používání **správce hesel**. Tato možnost funguje téměř univerzálně, zvyšuje bezpečnost vašich hesel a činí přihlašovací proceduru pohodlnější.

Dobří správci hesel ještě navíc vaše data důkladně šifrují. I kdyby tedy útočník získal přístup k datovému souboru, musí jej nejprve dešifrovat, než z něj získá užitečné informace. Za normálních okolností je to však velmi obtížné a většinou to nestojí za námahu. V porovnání s alternativními možnostmi vám pravidelné používání správce hesel zajistí, že budete mít vždy k dispozici seznam účtů, který můžete podle potřeby aktualizovat.

Pokud jsou navíc vaše citlivá osobní data uložena lokálně, nikoli v cloudu, útočníky to odradí. Získání údajů jedné osoby nebo rodiny pak totiž vyžaduje značné úsilí. U dobrých správců hesel si také můžete vybrat, kde budou vaše osobní údaje uloženy.

Mnoho lidí však používá vlastní systémy a často mají tendenci sdílet společné věci včetně společných hesel. Tyto praktiky jsou útočníkům velmi dobře známé. Tato skutečnost je zapracována do hackerských algoritmů a v důsledku toho mohou být vlastní systémy pro správu hesel spíše škodlivé než prospěšné. Předpokládejme, že váš systém pro správu hesel je vynikající. I tak zde existuje potenciální riziko. Pokud útoky na další systémy/služby odhalí vaše heslo, zvyšuje se pravděpodobnost napadnutí i vašeho systému. Jakmile bude váš systém prolomen, může jej útočník použít k odhalení vašich přihlašovacích údajů i na dalších webových stránkách. A konečně, používání vašeho systému namísto regulérního správce hesel je tak ve skutečnosti pomalejší a mnohem méně výhodné.

Dlouhé přístupové fráze (slovní spojení) jsou často vhodnější než hesla. Mnoho webových stránek je však nepodporuje (omezení počtu znaků, tj. délky přístupové fráze). Ačkoli jsou snadněji zapamatovatelné než hesla, neřeší problém lidí, kteří používají totožná hesla na různých webech nebo se spoléhají na systém.

Jako hlavní heslo pro správce hesel, které je klíčem k odemčení všech údajů o heslech, se však doporučuje používat přístupovou frázi. Ta vám umožní snadno zvolit a zapamatovat si extrémně dlouhé hlavní heslo.

Pokud uvažujete o ukládání hesel přímo do prohlížeče, existuje řada důvodů, proč je toto řešení nedůvěryhodné. Prohlížeče totiž neberou zabezpečení hesel až tak vážně, jak by měly, vzhledem k tomu, že pro přístup k uloženým údajům nevyžadují hlavní heslo. Stačí, abyste byli přihlášeni k počítači. Při využívání jiných, než vlastních počítačů je tato metoda i značně nepohodlná. Hesla prohlížeče jsou totiž využitelná pouze v rámci konkrétního prohlížeče. To v dnešní době není praktické, protože vaše heslo je vyžadováno nejen pro webové, ale často i pro mobilní aplikace.

Správci hesel umožňují ukládat více než jen jednoduchá hesla, což je poměrně užitečné. Správce hesel můžete použít k ukládání a doplňování pověření, například kreditních karet, které můžete následně použít i v prohlížečích. Ve správci hesel můžete uchovávat i celou řadu jiných citlivých údajů, například licence, osobní údaje, čísla bankovních účtů a další informace.

Považujte proto správce hesel za digitální trezor, který můžete nosit stále s sebou.

VÝHODY

Správci hesel navíc nabízejí i další výhody související s jejich praktickým využitím:

- Jsou integrováni do oblastí, ve kterých je nutné používat hesla, přičemž vytváření, aktualizace a vyplňování hesel je rychlé a jednoduché.
- Jsou kompatibilní s mnoha platformami a hesla mohou být volitelně synchronizována.
- Fungují dobře za různých provozních podmínek a v různých variantách, obvykle se i snadno udržují.
- Bezpečnost je u nich založena na robustním End-to-End šifrování, které zajišťuje bezpečnost dat i v případě, kdy dojde k jejich narušení.

Aby byl správce hesel účinný, musíte mít na paměti několik klíčových bodů:

- Měl by se používat na všech vašich webových stránkách, všude a bez výjimky. Výjimky pouze zvyšují zranitelnost a složitost systému (tj. snižují pravděpodobnost úspěšné obrany).
- Pro každý web vytvořte jedinečné heslo. Pokud máte tu možnost, vytvořte ho co nejdelší. Mělo by mít minimálně 20 až 30 znaků. Čím je heslo delší, tím je složitější ho prolomit (náročnost roste exponenciálně s délkou hesla). Vzhledem k tomu, že správce hesel je většinou vyplní za vás, nebudete je muset zadávat ručně.
- Zdaleka ne všechny webové stránky plně podporují správce hesel, hesla je proto občas nutné před vložením do přihlašovacího formuláře zkopírovat do schránky. To je však velice riskantní, protože tak snadno může dojít k odhalení vašeho hesla. Většina správců hesel proto po krátké době automaticky odstraní hesla ze schránky.
- Některé služby vyžadují často nesmyslné požadavky ohledně struktury hesla, například aby byla dlouhé alespoň 12 znaků. Zde lze s výhodu využít správce hesel, protože tato hesla generují zcela náhodně, což je asi nejbezpečnějším řešením, jakého lze vzhledem k omezením získat.
- Svoje hlavní heslo vytvořte co nejdelší a používejte jen takové heslo, které lze těžko uhodnout. Hlavní heslo je dobré pravidelně měnit, abyste snížili nebezpečí jeho úniku nebo zachycení prostřednictvím spyware, který tajně zaznamenává stisky kláves, tzv. **Keylogger**.
- Pokud si chcete vyměňovat hesla, nechte druhou osobu vytvořit vlastní trezor a pomocí funkce sdílení ve správci hesel s ní vybraná hesla sdílejte.

Mezi oblíbené správce hesel patří:

- LastPass
- Dashlane
- LogMeOnce

- 1Password
- Keeper
- KeePass

Některé z výše uvedených správců hesel jsou zdarma, za některé musíte zaplatit. U některých správců hesel jsou zdarma jen základní funkce, za pokročilé funkce je nutné si připlatit.

Interaktivní prvek

4.3 Dvoufázová autentifikace (2FA)

DEFINICE

Dvoufázová autentifikace 2FA je další vrstvou zabezpečení, která ověřuje, že každý, kdo se aktuálně pokouší získat přístup k online účtu, je tím, za koho se vydává.

Uživatel musí nejprve zadat své *přístupové údaje* (uživatelské jméno a heslo). Následně je vyzván k odeslání dalších doplňujících údajů tak, aby mohl získat plnohodnotný přístup.

Vhodným příkladem dvoufázového ověření je výběr peněz z bankomatu. Požadovanou transakci lze dokončit pouze se správnou kombinací bankovní karty (prostředku, který vlastníte) a kódu PIN (osobní identifikační číslo, tj. prostředek, který znáte).

Většina webových stránek dnes nabízí možnost ověření prostřednictvím SMS. Pro účely dvoufázového ověření 2FA se však stále více uplatňují mobilní zařízení.

VÝHODY

Výhody jsou jednoznačné:

- Využitím mobilních zařízení, která obvykle máme stále při sobě, již nejsou zapotřebí žádné další ověřovací tokeny.
- Dynamicky generované přístupové kódy jsou daleko bezpečnější než klasické pevné (statické) přihlašovací údaje, jelikož se neustále mění.

NEVÝHODY

Existují však i jisté nevýhody:

- Diskomfort – kdykoli je vyžadováno ověření, musí mít uživatel nabitý mobilní telefon a být v dosahu mobilní sítě. Přístup je často nemožný bez aktivace určitých záložních řešení, např. pokud telefon není schopen zobrazit příchozí zprávy (pokud je poškozený nebo se vypne z důvodu aktualizace nebo vlivem extrémních podmínek (např. v zimě)). Je také obvyklé, že textové zprávy nedorazí okamžitě, což může způsobit další zpoždění v procesu ověřování kvůli kopírování nebo ručnímu vkládání vyžadovaných údajů.

Je třeba si uvědomit, že SMS zprávy nejsou zdaleka tak bezpečné, jak byste možná očekávali. Přenos SMS zpráv na mobilní zařízení je nezabezpečený a náchylný k odposlechu. Díky tomu mohou třetí strany zaslaný token odcizit a případně zneužít. Při obnově účtu je 2FA na mobilním telefonu často obcházeno. Moderní chytré telefony se používají ke kontrole e-mailů a přijmu, resp. odesílání textových zpráv. Velmi často býváte na svůj e-mail permanentně přihlášení. Telefon tak může bez omezení přijímat druhý ověřovací údaj. V případě jeho ztráty nebo krádeže mohou být snadno a úspěšně napadnuté všechny účty, pro které nastavena autentifikace e-mailem. V konečném důsledku inteligentní telefony tedy integrují do jednoho fyzického zařízení oba ověřovací faktory najednou. Pokud je uživateli telefon odcizen, může

zločinec získat přístup k jeho účtům. Hackeři mohou získat přístup k mobilním telefonním sítím i prostřednictvím klonování **SIM** (***Subscriber Identity Module***) karet. Pokud zařízení nepodporuje zprávy SMS, je dvoufázová autentifikace prostřednictvím hlasového hovoru prakticky jedinou možností její realizace.

Mobilní aplikace Authy

Předpokládejme, že máte chytrý telefon nebo jiné mobilní zařízení. V takovém případě můžete získat svůj dvoufázový autentifikační kód bez použití SMS nebo hlasového hovoru stažením a instalací jedné z mnoha populárních aplikací pro dvoufázovou autentifikaci přímo ve vašem zařízení. Jedná se o mnohem bezpečnější způsob přihlašování pomocí dvoufázové autentifikace. Aplikace jako Authy nebo Google Authenticator vytvářejí tzv. **TOTP** (***Time-based One-Time Passcode***) přímo v aplikaci.

VÝHODY

I kdyby se útočníkovi podařilo přesvědčit vašeho poskytovatele mobilních služeb, aby provedl výměnu SIM karty, stejně by neměl přístup k vašim ověřovacím kódům. Informace potřebné k vytvoření těchto kódů jsou uloženy ve vašem vlastním zařízení, nikoli na kartě SIM.

Po instalaci programu Authy do vašeho telefonu budete chtít nastavit svoje první účty ověřené 2FA. To se provede naskenováním **QR** (***Quick Response***) kódu (poskytnutého webem, na kterém si chcete zabezpečit účet) pomocí aplikace. Je velmi pravděpodobné, že po načtení počátečního kódu a ochraně prvního účtu začnete stejným způsobem chránit i své další účty.

Obr. 20. Skenování QR kódu v mobilní aplikaci Authy

Nyní si ještě musíte vybrat mezi uchováváním všech tokenů 2FA v jednom zařízení, případně jejich zálohování v cloudu.

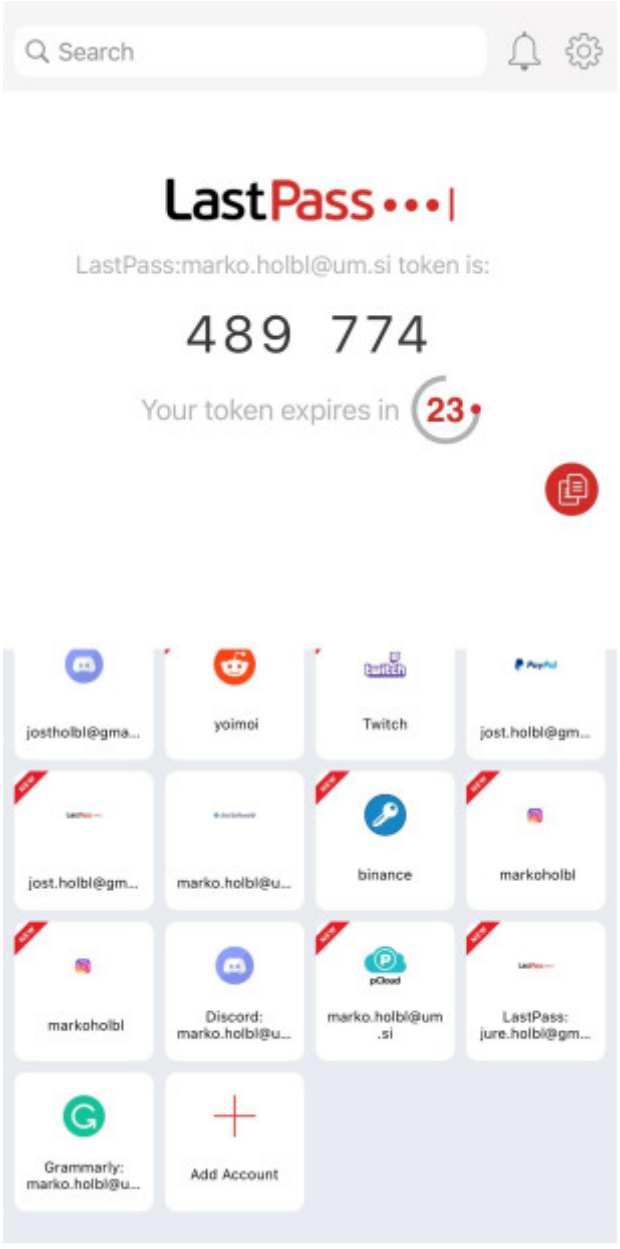
NEVÝHODY

Pokud si zvolíte pouze první možnost a následně např. zařízení ztratíte, aktualizujete nebo vám ho někdo odcizí, budete muset zajistit u všech služeb, u kterých jste si aktivovali 2FA, aby ji vypnuli. Po výměně telefonu se budete muset vrátit do svého účtu a ručně 2FA u každé jednotlivé služby znovu aktivovat.

VÝHODY

Proto vám Authy umožňuje zálohovat tokeny 2FA do svého bezpečného cloudového úložiště, které je přístupné pouze vám, takže můžete své účty vždy jednoduše obnovit, v případě ztráty, krádeže nebo výměny zastaralého zařízení.

Při zálohování tokenů 2FA do cloudu budete vyzváni k nastavení záložního hesla, které se použije k zašifrování vašich dat a následné synchronizaci v rámci použité cloudové služby. Vaše osobní údaje jsou v této cloudové platformě v mimořádném bezpečí, protože vaše heslo se nikde fyzicky neukládá – je však nesmírně důležité, abyste si ho sami pamatovali.



Obr. 21. Používání mobilní aplikace Authy

Následně je možné nainstalovat Authy i na ostatní zařízení. Aplikace automaticky synchronizuje tokeny u každého zařízení, kde je Authy nainstalováno, avšak pouze v případě, pokud jste je synchronizovali s cloudem Authy. V případě, že máte pouze jedno mobilní zařízení, můžete si také stáhnout aplikaci

Authy Desktop nezávislou na prohlížeči.

Interaktivní prvek

4.4 Aspekty bezpečného ukládání hesel (na straně serveru)

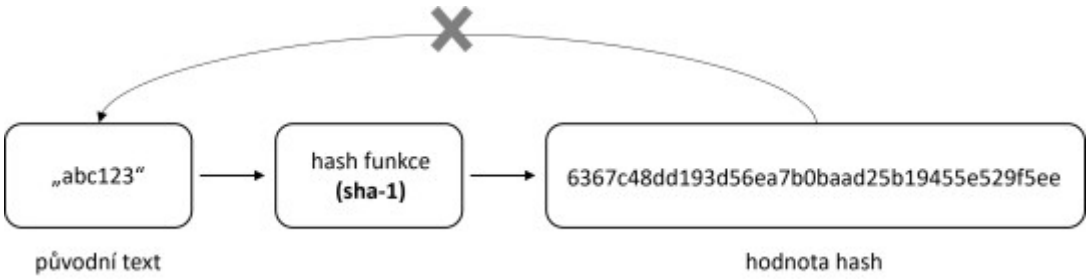
Hesla musí být dostatečně chráněna na straně ověřovatele (obvykle serveru). Denně je hlášeno mnoho případů narušení bezpečnosti dat, a proto se nelze spoléhat výhradně na bezpečnost systémů ověřovatele. Hesla tedy nelze uchovávat jako prostý text (tj. v otevřeném formátu) a měla by být uložena bezpečným způsobem. Nejprve si však stručně představíme některé pojmy potřebné pro pochopení bezpečného ukládání hesel.

4.4.1 Ukládání hashovaných hesel

DEFINICE

Kryptografická hashovací funkce přijímá vstupní údaj (nebo zprávu) a vrací alfanumerický řetězec pevné délky.

Tento řetězec je známý jako hodnota HASH, fragment zprávy, digitální otisk, souhrn (digest) nebo kontrolní součet.



Obr. 22. Jak funguje hashování

Obrázek 22 znázorňuje proces hashování. Začneme slovem „abc123“ a pomocí hashovací funkce **SHA-1** (***Secure Hash Algorithm 1***) získáme alfanumerický výstup pevné délky, kterému říkáme hash nebo hashová hodnota. Pomocí této hodnoty hash nejsme schopni obnovit náš původní vstupný text. Hodnotu hash nelze ani invertovat a poté ji využít ke zjištění původního obsahu, protože hashovací funkce jsou jednosměrné, a tudíž nereverzibilní. Pokud projde stejný materiál stejnou hashovací funkcí, měl by být výsledek opět stejný. Takže místo ukládání hesla ve formátu prostého textu jej můžeme hashovat pomocí hashovací funkce a uložit jeho hashovací hodnotu.

user_name	heslo
john	abc123
sam	abc123
alice	xyz456

→

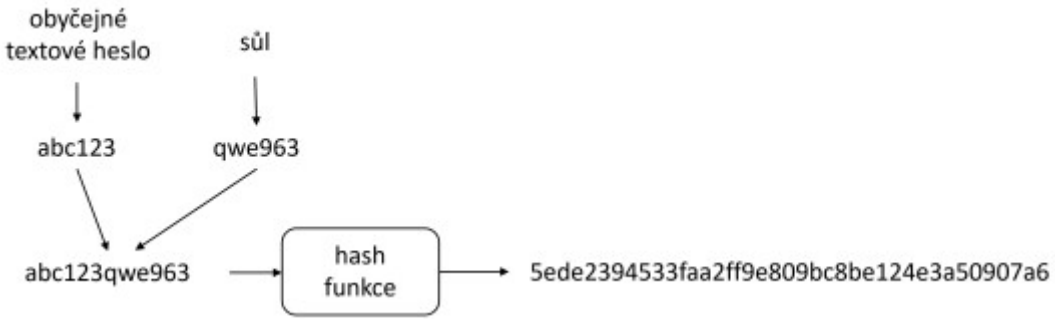
user_name	hash heslo
john	6367c48dd193d56ea7b0baad25b19455e529f5ee
sam	6367c48dd193d56ea7b0baad25b19455e529f5ee
alice	0772dbe339a885eb2ed73c1fe842d2ef6e9003a3

Obr. 23. Ochrana uložených hesel pomocí hashování

Pokud se uživatel pokusí přihlásit do systému, použije se hashovací funkce k hashování uživatelova hesla a výsledek se následně porovná s hodnotou hash uloženou v tabulce. Pokud jsou obě hodnoty hash stejné, bude uživateli povoleno přihlášení a vstup do systému. Na obrázku 23 mají john a sam stejné heslo „abc123“ a jejich hodnoty hash jsou po použití hashovacího algoritmu taky stejné. Uvažujme případ, kdy má john přístup do databáze a může vidět hash hesla. Pak může john vidět, že hodnota hash jeho hesla je stejná jako hodnota hash hesla samova. Výsledkem je, že john bude moci použít přihlašovací údaje Sama k přihlášení do systému. Abychom se této situaci vyhnuli, můžeme použít techniku nazývanou **solení** (*salting*).

4.4.2 Hashování technikou solení

Naším cílem je pomocí hashování technikou solení (tj. náhodným řetězcem) vytvořit hash hodnotu hesla jedinečnou. Systém proto generuje náhodnou posloupnost znaků nazývanou „*sůl*“. V okamžiku, kdy uživatel zadá heslo ve formátu obyčejného textu (*plain text*), připojí se k němu vytvořená náhodná sada znaků. Následně je pomocí hashovací funkce získána hodnota hashe z vloženého textu (tzv. ***solený hash***). V tomto případě je třeba uložit hodnotu soli každého uživatele.



Obr. 24. Proces hashování technikou solení

I když mají john a sam stejné heslo, jejich hodnoty soleného hashe se liší (viz obrázek 25).

Interaktivní prvek

Během přihlašovacího procesu systém načte z databáze příslušnou hodnotu soli uživatele, připojí ji ke vstupnímu heslu, aplikuje hashovací funkci a porovná výslednou hodnotu hash s hodnotou hash zaznamenanou v tabulce. Pokud se obě hodnoty hash shodují, je uživatel úspěšně ověřen.

user_name	hodnota soli	solené hash heslo
john	qwe963	5ede2394533faa2ff9e809bc8be124e3a50907a6
sam	hjk521	6367c48dd193d56ea7b0baad25b19455e529f5ee
alice	asd753	0772dbe339a885eb2ed73c1fe842d2ef6e9003a3

Obr. 25. Příklad tabulky ukládající solená a hashovaná hesla

Minimální ochrana uložených hesel by měla zahrnovat hashování technikou solení. Kromě toho NIST navrhuje:

- zablokovat uživatele v případě, že příliš často použije nesprávné heslo (např. po třech neúspěšných pokusech se uživatel nesmí po dobu jedné minuty pokusit o další přihlášení),
- povolit v heslech emotikony, znaky ASCII a Unicode
- a povolit funkce kopírování a vkládání (*Copy & Paste*) u polí pro hesla, aby bylo používání správců hesel a vícefázového ověřování pohodlnější.

Interaktivní prvek

KAPITOLA 5

Autentifikace bez hesla

S ohledem na všechny problémy a slabiny hesel není myšlenka ověřování bez hesel nová. Jak již název napovídá, ověřování bez hesla umožňuje uživateli přihlásit se nebo získat přístup bez zadání hesla nebo zodpovězení bezpečnostních otázek. Autentifikace bez hesla eliminuje potřebu zadávat potenciálně problematická hesla a řešit jejich správu a zároveň zvyšuje bezpečnost uživatelských účtů tím, že snižuje jejich zranitelnost vůči útokům. Existují různé mechanismy autentifikace bez hesla, jako např. bezdotykové přístupové karty, fyzické tokeny, zařízení USB/klíče, magické odkazy, biometrické rozpoznávání, mobilní aplikace atd. Většina těchto technik se dnes již běžně používá např. při vícefázovém ověřování s cílem zvýšit úroveň zabezpečení. Některá z těchto řešení však mohou být použita jako systém autentizace prvního stupně.

Prvky užívané pro ověření uživatele bez hesla lze obvykle rozdělit do dvou kategorií:

- Příklady **prvků vlastnictví** jsou chytré telefony, tokeny OTP, čipové karty nebo hardwarové tokeny (tedy „něco, co uživatel má“).
- Příklady **základních (vrozených) faktorů** jsou otisky prstů, skeny sítnice, rozpoznávání obličeje nebo hlasu a další biometrické identifikátory (tedy „něco, čím uživatel je“).

Autentifikace bez hesla se často zaměňuje s vícefázovým ověřením MFA, protože obě používají různé faktory ověřování, případně jejich kombinace. Zatímco MFA se používá jako další bezpečnostní vrstva bezprostředně navazující na ověřování založené na hesle, ověřování bez hesla nevyžaduje zapamatování si tajného údaje a k ověření identity obvykle používá pouze jeden vysoce bezpečný faktor, což je pro uživatele rychlejší a jednodušší.

Obecně lze říct, že hesla se obtížně pamatují a požadavky na ně se stále mění a zpřísňují. Různé weby mohou používat odlišné zásady pro zadávání hesel, takže heslo vygenerované pro jeden web nemusí fungovat na jiném a opačně. Zapamatovat si heslo vygenerované podle moderních rozšířených zásad je často velmi náročné.

Stejně jako v případě **FIDO** (***Fast Identity Online***) se zde uplatňují různé standardy. Ačkoli ověřování bez hesla a technologie FIDO již nějakou dobu existují, online služby a poskytovatelé identit je zatím ve velkém měřítku nepoužívají. Ověřování bez hesla se stane budoucností ověřování díky zabudování biometrických funkcí do většiny moderních mobilních zařízení a notebooků.

VÝHODY

Autentifikace bez hesla vylepšuje komfort koncového uživatele tím, že zcela odstraňuje často komplikovaný proces zadávání hesla včetně jeho pamatování. Uživatel již nemusí vytvářet delší

a bezpečnější heslo, a přitom může získat jednotný přístup ke všem systémový prostředkům jednoduchým připojením USB zařízení/klíče nebo pouhým naskenováním otisku prstu.

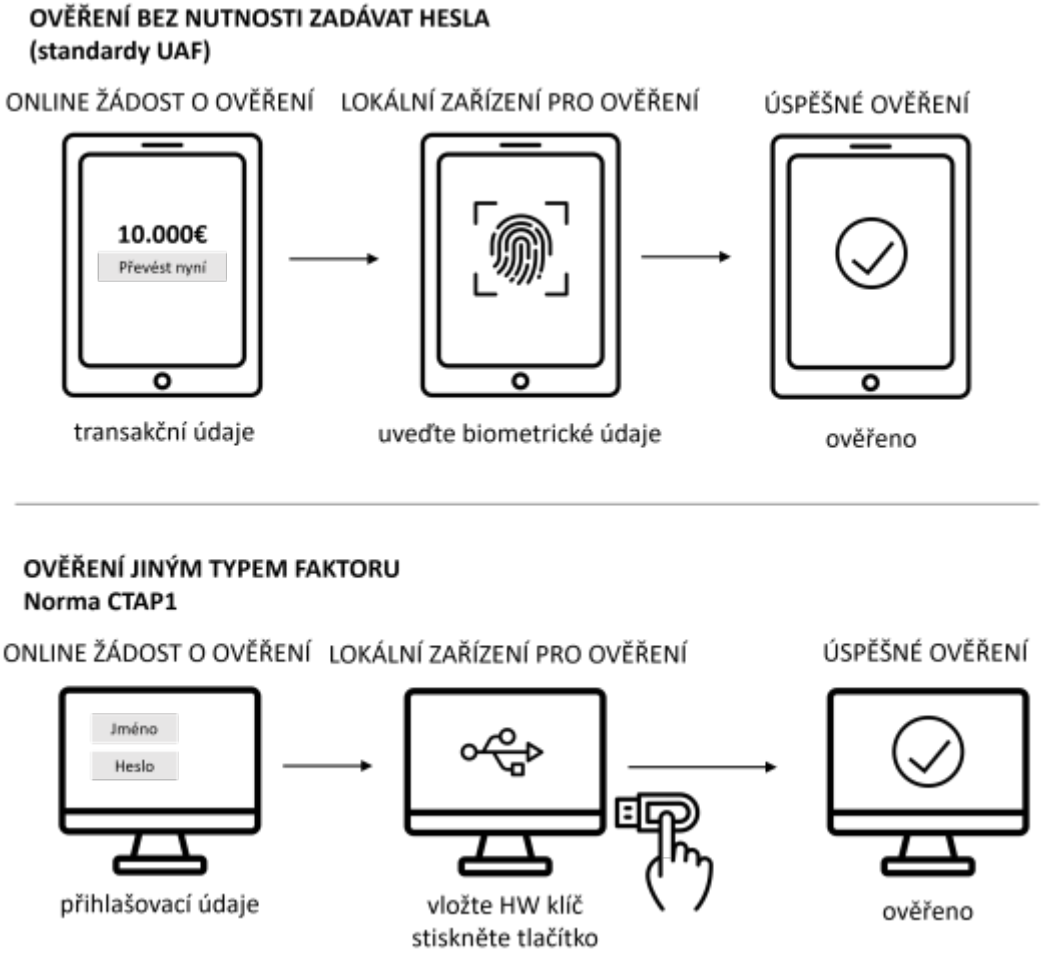
5.1.1 FIDO (Fast Identity Online)

FIDO je sada otevřených autentifikačních protokolů vytvořených aliancí FIDO Alliance umožňujících přihlášení bez hesla. Protokoly FIDO používají k bezpečnému ověřování základní kryptografické algoritmy s veřejným klíčem. Soukromé klíče nikdy neopustí zabezpečovací zařízení a veškerá komunikace je šifrovaná.

Obr. 26. Příklad autentifikace založené na FIDO

Aliance FIDO vydala tři sady standardů:

- **UAF** (***Universal Authentication Framework***): Autentizace bez hesla je součástí protokolu FIDO UAF. Uživatelé, kteří používají tento protokol, by měli reagovat na výzvu poskytnutou serverem FIDO a pomocí jednoho nebo více bezpečnostních faktorů dostupných v jejich zabezpečovacím/ digitálním zařízení ověřit svou identitu.
- **U2F** (***Universal Second Factor***): Funkci dvoufázové autentifikace poskytuje protokol FIDO U2F. Uživatelé musí k prokázání své totožnosti předložit dva identifikátory (faktory). Se zavedením protokolu FIDO2 byl tento protokol přejmenován na CTAP1.
- FIDO2: Nejnovější soubor specifikací aliance FIDO je známý jako FIDO2.



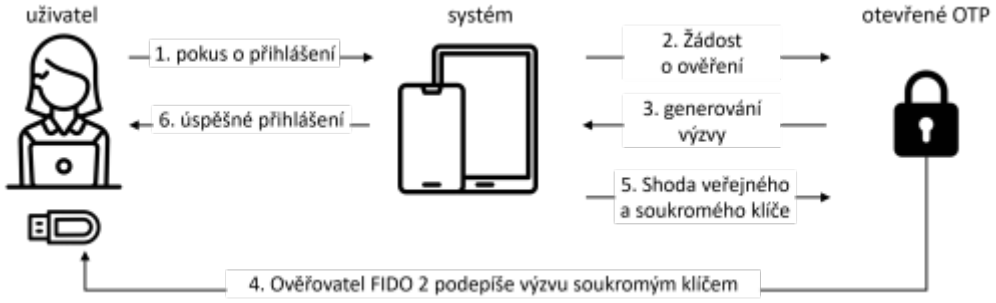
Obr. 27. Standardy UAF a U2F (CTAP1) pro autentifikaci bez hesla

5.1.2 FIDO2 a WebAuthn

Specifikace FIDO2 se skládá:

- ze standardu W3C WebAuthn (*Web Authentication*) a
- z protokolu FIDO **CTAP2** (***Client to Authenticator Protocol 2***).

FIDO2 umožňuje uživatelům využít běžná zařízení, mobilní i desktopová, ke snadné autentifikaci internetových služeb. WebAuthn je standardizované online **API** (***Application Programming Interface***) rozhraní pro autentifikaci pomocí protokolu FIDO, které je součástí mnoha platforem a prohlížečů. CTAP2 je verze CTAP, která uživatelům umožňuje využívat externí i interní (vestavěné) prostředky pro ověření a nabízí bez heslovou, dvojfázovou nebo vícefázovou autentifikaci. WebAuthn API je nástroj pro vytváření a správu pověření založených na principu veřejného klíče. Přehled autentifikace FIDO2 je uveden na obrázku 28.



Obr. 28. Autentifikace pomocí protokolu FIDO2

Interaktivní prvek

KAPITOLA 6

Úvod do problematiky digitálního podpisu

DEFINICE

Digitální podpis je matematický systém na kontrolu validity (platnosti, pravosti, správnosti) digitálních zpráv nebo dokumentů.

Věrohodný digitální podpis dává příjemci spolehlivou zprávu o tom, že danou zprávu vytvořil důvěrně známý odesílatel (*autenticita*), a že nebyla při přenosu pozměněna, pokud jsou splněny nezbytné předpoklady (*integrita*).

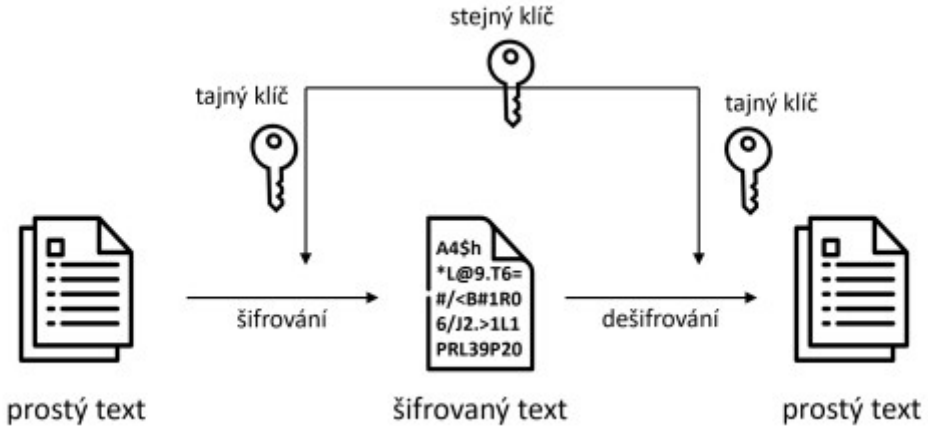
Hlavní cíle, kterých se digitální podpis snaží dosáhnout, jsou:

- **Autenticita**: Digitální podpisy jsou vázané na konkrétního uživatele prostřednictvím jeho soukromého klíče. Díky tomu je možné určit, kdo vlastní soukromý klíč použitý k podpisu původních dat/zprávy (např. dokumentu, e-mailu nebo souboru). Více informací o soukromých a veřejných klíčích naleznete níže.
- **Integrita**: Digitální podpisy využívají techniku hashování, která zajišťuje, že s obsahem zprávy nebude během přenosu manipulováno, tj. zpráva nebude zfalšována / pozměněna. Více informací o hashování naleznete níže.

Digitální podpis je tedy jedním ze způsobů ověřování identity subjektu (entity), ale nejprve je nutné objasnit některé pojmy, aby bylo možné ukázat, jak lze digitální podpis při ověřování použít.

6.1 Kryptografie založená na veřejném klíči

Abychom pochopili fungování digitálního podpisu, je třeba nejprve vysvětlit princip asymetrické kryptografie, často označované jako *kryptografie s veřejným klíčem*. Na rozdíl od klasického (symetrického) šifrování, které k šifrování používá pouze jeden klíč, **asymetrické šifrování** používá dvojici klíčů. **Šifrování** je proces kódování informací, jak je znázorněno na obrázku 29.



Obr. 29. Symetrické šifrování

Představte si, že chcete někomu poslat šifrovanou zprávu pomocí klasického šifrování. V tomto případě se obě strany musí dohodnout na jediném klíči. Klíč však v tomto případě nelze navzájem bezpečně přeposlat, protože pokud by ho zachytil kdokoliv neoprávněný, mohl by následně sledovat veškerou vaši komunikaci, resp. obsah vašich zpráv.

DEFINICE

Asymetrické šifrování naopak využívá dvojici klíčů, *veřejný a soukromý klíč*, které k sobě matematicky patří. Pouze správně spárovaný soukromý klíč může dešifrovat to, co je zašifrováno veřejným klíčem.

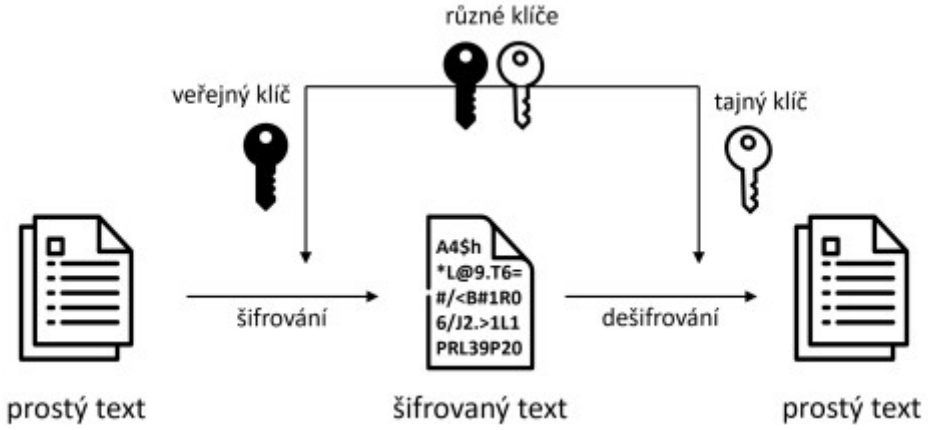
Pokud tedy někdo požaduje, aby mu ostatní posílali zašifrované zprávy, stačí, když jednoduše zveřejní svůj veřejný klíč, aby ho všichni ostatní mohli následně použít. Poté lze jednoduše použít soukromý klíč pro dešifrování zpráv zašifrovaných veřejným klíčem, protože zprávy zašifrované veřejným klíčem lze dešifrovat pouze soukromým klíčem. To je velmi užitečné, neboť se v tomto případě nemusíme starat o bezpečné sdílení veřejného klíče.

Stručně řečeno, aby mohly obě strany bezpečně komunikovat prostřednictvím asymetrického šifrování, musí tento proces probíhat následujícím způsobem:

- Obě strany si navzájem vymění veřejné klíče.
- *Osoba 1* zašifruje zprávu, kterou chce odeslat pomocí veřejného klíče *osoby 2* a následně ji odešle *osobě 2*.

- *Osoba 2* dešifruje přijatou zprávu pomocí svého soukromého klíče.

Tento proces je znázorněný na obrázku 30.



Obr. 30. Proces asymetrického šifrování založený na principu veřejného klíče

DEFINICE

Digitální podpisy tedy fungují tak, že cokoliv se podepíše (zašifruje) soukromým klíčem, se následně ověří veřejným klíčem, který je s ním spárovaný. Takže jednotlivé klíče z daného páru klíčů se pak používají proti sobě navzájem.

Důvodem je, že podepisující osoba je jedinou osobou, která má přístup ke svému soukromému klíči použitému k podpisu. Proto si můžete být naprosto jisti, že zprávu/dokument podepsala výhradně, a právě tato osoba. Kdokoli pak může pomocí veřejného klíče ověřit (tj. úspěšně dešifrovat zprávu), že zprávu vytvořil vlastník veřejného klíče.

Interaktivní prvek

6.2 Princip digitálního podpisu

Jak bylo uvedeno v předchozím textu, pro digitální podpis se používá dvojice kryptografických klíčů, která se skládá z veřejného a soukromého klíče. Páry kryptografických klíčů se používají k šifrování (zamykání) a dešifrování (odemykání) zdrojových dat stejným způsobem, jako se k zamykání a odemykání používají fyzické klíče. Soukromé klíče jsou zabezpečeny a jsou tedy důvěrné, protože pokud se někdo dozví soukromý klíč jiné osoby, může podepsat zdrojová data za tuto osobu. Na druhou stranu se předpokládá, že veřejné klíče budou sdíleny s kýmkoliv. Data zašifrovaná soukromým klíčem lze dešifrovat pouze veřejným klíčem, čímž se zpřístupní původní data/údaje.

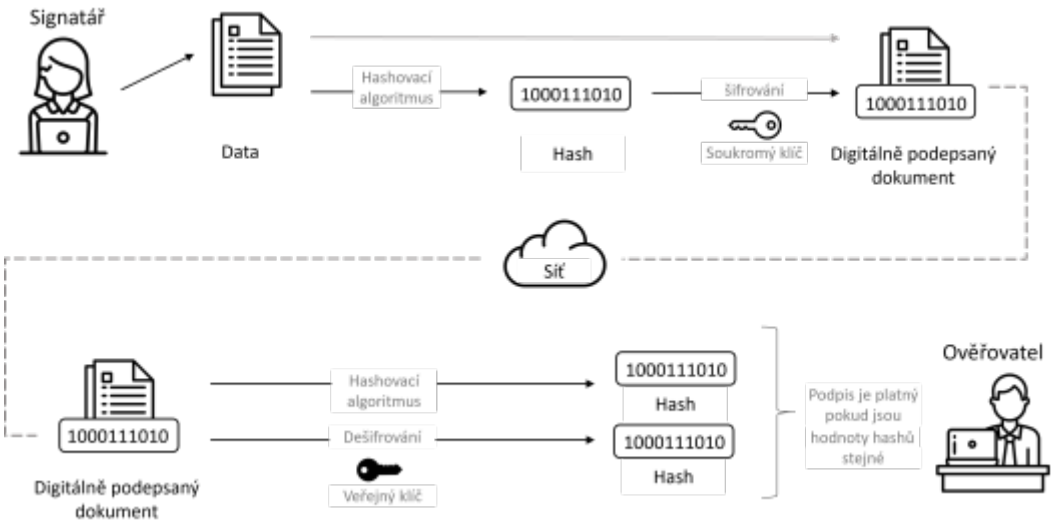
DEFINICE

Při vytváření digitálního podpisu se využívá asymetrická kryptografie a hashovací funkce.

Tyto dva stavební bloky jsou zakomponovány do vlastního procesu digitálního podpisu následovně:

1. Hashováním odesílatel vypočítá hash zdrojového obsahu, který chce doručit.
2. Odesílatel následně vypočtený hash zašifruje svým soukromým klíčem a vytvoří digitální podpis.
3. Poté lze obsah i digitální podpis odeslat příjemci.
4. Po přijetí zprávy příjemcem použije příjemce veřejně dostupný veřejný klíč odesílatele k dešifrování zašifrovaného digitálního podpisu odesílatele. Pokud se to podaří, potvrdí se identita odesílatele jako vlastníka soukromého klíče použitého k zašifrování souboru.
5. Příjemce z přijaté zprávy získá její původní obsah a vygeneruje hash tohoto obsahu.
6. Obsah je potvrzen identickým, tj. totožný s tím, který poskytl odesílatel, pokud se vypočtený hash příjemce shoduje s hashem odesílatele. Pokud se hashe neshodují, došlo k manipulaci s obsahem zprávy, a v tom případě není digitální podpis platným.

Grafické znázornění celého procesu je na obrázku 31.



Obr. 31. Schematické znázornění vytvoření a ověření digitálního podpisu

Protože je veřejný klíč odesílatele veřejně dostupný, může kdokoli dešifrovat zašifrovaný obsah, který odesílatel odesílá. V důsledku toho tato metoda šifrování ověřuje pouze integritu, nikoli důvěrnost.

Někdo by se mohl ptát, proč před vlastním podpisem dat generujeme jejich hashovanou hodnotu? Jednoduše je to proto, že hash signatura digitálního podpisu je mnohem menší. Proces vytváření a ověřování digitálního podpisu se tím také zrychlí, jelikož se porovnávají pouze hashované hodnoty, a nikoliv celá data nebo dokument. Všimněte si, že to skutečně funguje, protože hashovací algoritmy vždy vytvářejí hodnotu určité, resp. jednotné délky.

VÝHODY

Jak už asi tušíte, digitální podpis poskytuje několik výhod, včetně několika následujících:

- zvyšuje bezpečnost a důvěryhodnost, protože ho nelze zpětně napodobit nebo zfalšovat;
- zajišťuje nezpochybnitelnost zdrojových dat šifrujícímu subjektu;
- zabezpečuje integritu přenášených dat, resp. údajů.

NEVÝHODY

Digitální podpis však má i své nevýhody, např.:

- skutečnost, že neexistuje žádný způsob, jak podpis (důvěryhodnost zdrojových dat) po jeho doručení odvolat, čímž se stává nezvratným;
- použití veřejných klíčů znemožňuje utajení, tedy kdokoli, kdo disponuje veřejným klíčem, může daný podpis ověřit.

Interaktivní prvek

Interaktivní prvek

KAPITOLA 7

Infrastruktura veřejného klíče

Již jsme se seznámili s pojmy digitální podpis a kryptografie s veřejným klíčem. Ukazuje se však, že pro správné fungování celého konceptu v reálné aplikaci potřebujeme ještě něco navíc. Potřebujeme něco, co se nazývá infrastruktura veřejného klíče **PKI** (***Public Key Infrastructure***).

DEFINICE

PKI je soubor technologií, procesů a entit, který umožňuje a zajišťuje bezpečnou komunikaci v nezabezpečených veřejných sítích.

Například infrastruktura PKI přidává postfix „S“ k protokolu HTTPS, a tedy, pokud si zobrazíte webovou stránku ve svém prohlížeči, pravděpodobně ho využíváte k tomu, abyste se ujistili, že její obsah pochází z důvěryhodného zdroje. Infrastruktura PKI umožňuje regulovaný přístup k systémům a zdrojům, ochranu dat a odpovědnost za transakce jednoznačným potvrzením identity osob, zařízení a služeb.

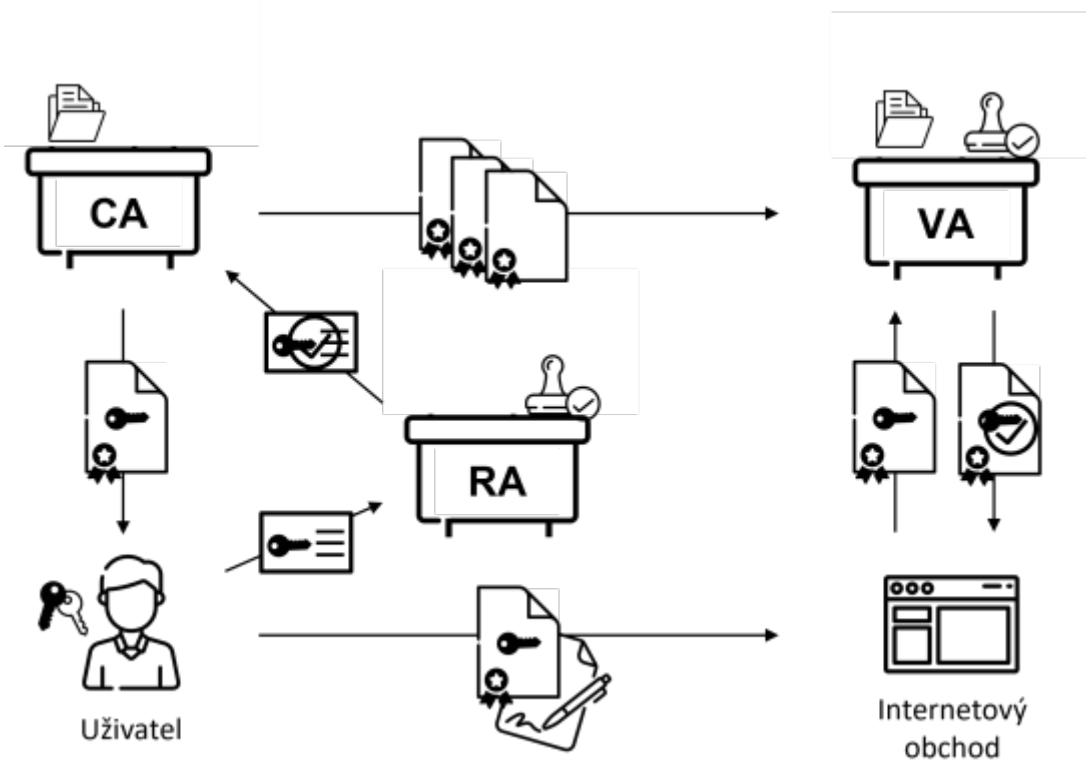
Infrastruktura PKI se využívá v celé řadě aplikací, včetně např. zabezpečení komunikace v internetu věcí **IoT** (***Internet of Things***) nebo digitálního podepisování dokumentů. Infrastruktura PKI založená na asymetrické kryptografii, se dnes také běžně používá k nastavení bezpečné elektronické komunikace, jako jsou např. online nákupy, bankovnictví a e-maily, případně komunikace mezi uživateli a webovými stránkami, ke kterým se připojují pomocí protokolu HTTPS. Infrastruktura PKI umožňuje silné ověřování, šifrování dat a digitální podpisy osob, služeb a dalších entit tím, že jim poskytuje tzv. digitální identitu. Tyto metody zabezpečení zajišťují bezpečný přístup k fyzickým a digitálním zdrojům, bezpečnou komunikaci mezi lidmi, službami a věcmi a digitální podepisování dokumentů, transakcí nebo jiných dat.

7.1 Prvky infrastruktury veřejného klíče

Infrastruktura PKI se skládá z následujících částí (prvků):

- certifikační autorita **CA** (***Certificate Authority***),
- registrační autorita **RA** (***Registration Authority***),
- validační autorita **VA** (***Validation Authority***),
- digitální certifikáty.

A samozřejmě kryptografie s veřejným klíčem **PKC** (***Public-Key Cryptography***).



Obr. 32. Prvky infrastruktury veřejného klíče PKI

Certifikační autorita

DEFINICE

Certifikační autorita je společnost, která vytváří a distribuuje digitální certifikáty.

Digitální certifikát potvrzuje, že subjekt, který je v certifikátu uveden, disponuje vlastním veřejným klíčem. Ověřovatelé pak mohou důvěřovat podpisům a tvrzením o soukromém klíči, který odpovídá certifikovanému veřejnému klíči. Certifikační autorita slouží jako důvěryhodná třetí strana, které důvěřuje jak subjekt (vlastník) certifikátu, tak strana, která se na certifikát spoléhá.

Podepisování certifikátů používaných v protokolu HTTPS, zabezpečeném protokolu pro surfování, je jedním z nejrozšířenějších použití certifikačních autorit. Další oblíbenou aplikací je vydávání průkazů totožnosti vládami jednotlivých států, které mohou být použity pro digitální podepisování nebo elektronickou veřejnou správu.

Registrační autorita

DEFINICE

V infrastrukturách veřejných klíčů zajišťuje **registrační autorita** registraci certifikátů. Odpovídá za přijímání žádostí o podepsání certifikátu od jednotlivců, serverů, věcí a dalších aplikací, a to ať už jde o prvotní zápis nebo jeho obnovení. Tyto žádosti jsou ověřovány registrační autoritou a následně předávány certifikační autoritě.

Registrační autorita je rovněž zodpovědná za správu životního cyklu certifikátu. Podívejme se na případ, kdy dojde k odvolání certifikátu. Registrační autorita používá obchodní logiku pro příjem žádostí, včetně metod pro ověření původu žadatele a strany, která by měla daný certifikát vlastnit.

Z důvodu zachování dostupnosti a bezpečnosti je registrační autorita obvykle oddělena od certifikační autority. K registrační autoritě lze přistupovat prostřednictvím uživatelsky přívětivého grafického rozhraní **GUI** (***Graphic User Interface***) nebo pomocí rozhraní API a standardních protokolů, které lze snadno integrovat.

Validační autorita

Certifikáty PKI ověřuje validační autorita. Příkladem služeb ověřování certifikátů je např. přístup k seznamům odvolaných certifikátů **CRL** (***Certificate Revocation Lists***), protokol **OCSP** (***Online Certificate Status Protocol***) a stahování řetězových certifikátů certifikačních autorit. Vzhledem k tomu, že certifikáty mohou být vydávány a rušeny, je nezbytné ověřit pravost certifikátu před tím, než mu začnete důvěřovat. Úkolem ověřovací autority je řešit právě tento aspekt.

Vydávající certifikační autorita je odpovědná za poskytování aktualizací stavu certifikátu ověřovací autoritě v souladu se stanovenou politikou. Použitím seznamů CRL CA se pak můžete spolehnout na to, že každá zapojená certifikační autorita zveřejní seznam odvolaných digitálních certifikátů.

Digitální certifikát

Digitální certifikát je forma elektronické identifikace jednotlivých subjektů nebo organizací, podobně jako občanský průkaz. Obsahuje informace, jako je identita, sériové číslo a datum (dobu) platnosti. Vedle těchto informací můžeme rovněž zobrazit digitální podpis certifikační autority, který zajišťuje pravost certifikátu, a veřejný klíč držitele certifikátu. Infrastruktura PKI umožňuje například ověřovat spojení a v kombinaci s dalšími kryptografickými přístupy zabezpečuje spojení mezi dvěma komunikujícími stroji, protože identitu obou stran lze potvrdit pomocí digitálních certifikátů. Téměř všechny dnes vydávané certifikáty odpovídají standardu ITU-T X.509.

Existuje mnoho typů certifikátů:

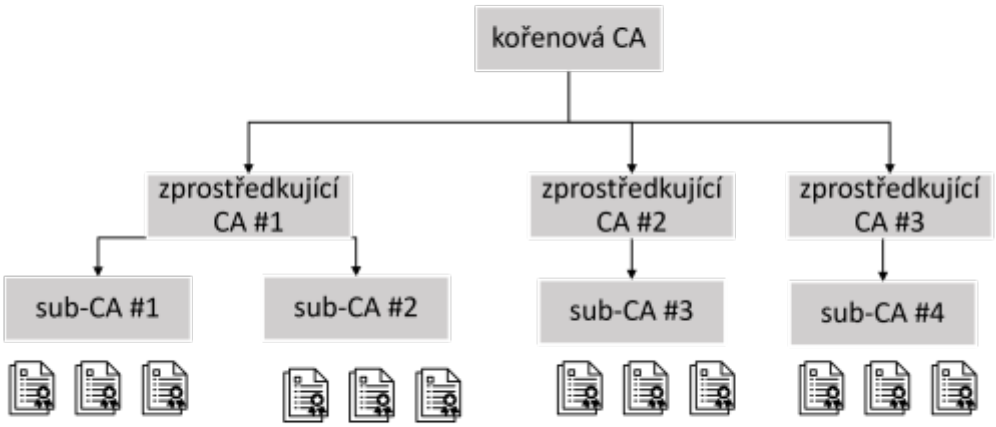
- **Certifikáty pro podepisování kódu**: Kód je ověřen jako kód pocházející od vývojářů, tzn. že nebyl pozměněn, což činí software důvěryhodným. Slouží také k podepisování softwarových verzí a ověřování softwaru od prodejce nebo vývojáře, čímž se prokazuje jeho legálnost.
- **E-mailové certifikáty**: Protokol **S/MIME** (***Secure/Multipurpose Internet Mail Extensions***) lze použít k zabezpečení a ověření e-mailů, což umožňuje odesílateli prokázat autorství a zabránit neoprávněné manipulaci.
- **Certifikáty na podepisování dokumentů**: Programy firem Adobe, Microsoft a dalších se používají k podepisování dokumentů tak, aby bylo zajištěno, že nebudou pozměňovány a budou důvěryhodné. Tento typ certifikátu je téměř vždy zobrazen u digitálního podpisu na dokumentu.
- **Certifikáty TLS (*Transport Layer Security*)**: Používají se pro zabezpečené připojení HTTPS.

Obr. 33. Příklad digitálního certifikátu v systému Microsoft Windows

Interaktivní prvek

7.2 Hierarchická struktura infrastruktury veřejných klíčů

V PKI je běžná hierarchie CA, které vytvářejí a vydávají digitální certifikáty nebo pověření. Lokální certifikační autority jsou oprávněny podepisovat digitální certifikáty pro zařízení od každé nadřazené certifikační autority. Koncová zařízení na nejnižším stupni hierarchie poskytují digitální certifikáty, které jsou povoleny nadřazenou lokální certifikační autoritou, která je vygenerovala a podepsala. Ty se někdy označují jako *certifikáty zařízení*. Lokální certifikační autority, které vytvářejí certifikáty zařízení, disponují vlastním certifikátem, který je autorizován digitálním podpisem jim nadřízené certifikační autority atd. PKI nakonec dospěje ke kořenovému certifikátu, který slouží jako základ této konkrétní domény ekosystému infrastruktury PKI.



Obr. 34. Příklad hierarchie PKI

Prostřednictvím hierarchického uspořádání ekosystémů PKI lze v případě vyzrazení nebo napadení soukromého klíče v ekosystému stanovit konkrétní úrovně pro odmítnutí přístupu nebo jeho úplné zablokování.

V závislosti na povaze narušení bezpečnosti může kdokoli zrušit certifikát jakéhokoli prvku PKI, od zařízení až po certifikační autoritu vyšší úrovně. Toto zneplatnění certifikátu navíc zneplatní i vše, co je v hierarchii pod daným prvkem.

Z toho tedy přímo vyplývá, proč jsou implementace PKI uspořádány do stromových hierarchií. Tato konstrukce totiž velmi snadno umožňuje vlastníkovi ekosystému provádět selektivní kontrolu škod

v případě jeho narušení. Z tohoto důvodu není vydávání certifikátů zařízení odvozených přímo od kořenové certifikační autority vhodným přístupem, neboť omezuje flexibilitu celého systému PKI. Pokud se totiž v tomto případě cokoliv pokazí, budeme nuceni zneplatnit a odvolat celý systém PKI a tím i všechna nasazená zařízení v terénu. Proto jsou certifikáty zařízení prakticky vždy vydávány lokálními certifikačními autoritami umístěnými v hierarchii pod danou kořenovou certifikační autoritou.

7.3 Životní cyklus digitálního certifikátu

Životní cyklus digitálního certifikátu začíná jeho vytvořením a lze jej stručně popsat následovně:

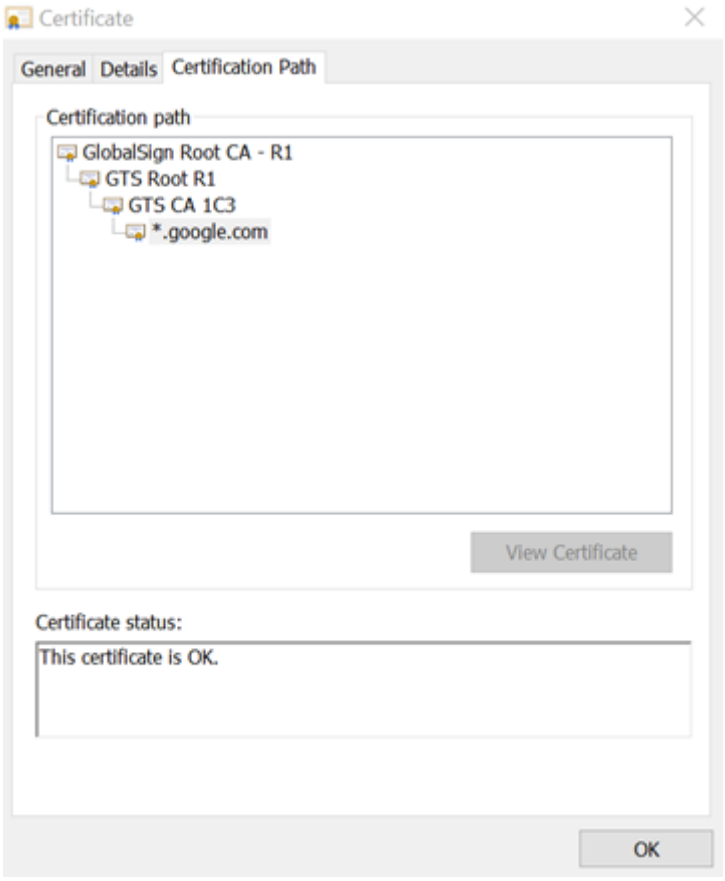
- **Registrace certifikátu**: Certifikační autorita obdrží od entity žádost o certifikát. Za entitu lze považovat osobu, zařízení nebo dokonce i jen několik řádků kódu.
- **Vydání certifikátu**: registrační autorita musí ověřit totožnost žadatele, což se obvykle provádí prostřednictvím přihlašovacích údajů nebo pomocí identifikace jinou registrační autoritou, která již totožnost žadatele ověřila.
- **Ověření certifikátu**: Server při každém použití digitálního certifikátu ověřuje u certifikační autority, zda je certifikát stále platný, zda jeho platnost nevypršela nebo nebyl zneplatněn.
- **Zneplatnění certifikátu**: Při prvním vydání certifikátů je vždy uveden i datum jejich platnosti. Po uplynutí tohoto data certifikační autorita zařadí certifikát na seznam odvolaných certifikátů CRL, což je forma černé listiny, která serveru říká, aby již těmto certifikátům nedůvěřoval.
- **Obnova certifikátu**: Certifikační autority lze nakonfigurovat tak, aby automaticky obnovovaly certifikáty po uplynutí doby jejich platnosti, i když obvykle vyžadují opětovné ověření identity.

Interaktivní prvek

Certifikační autority a řetězec důvěry

Pojem „řetězec důvěry“ odkazuje na vztah mezi digitálním certifikátem a důvěryhodnou certifikační autoritou. Aby byl certifikát důvěryhodný, musí být dohledatelný až k důvěryhodnému kořenovému certifikátu, od kterého byl vydán, což znamená, že všechny certifikáty v řetězci – serverový, zprostředkující i kořenový – musí být odpovídajícím způsobem důvěryhodné.

Na obrázku 35 vidíme, že pro server google.com je GTS-CA 1C3 certifikační autoritou nejnižší úrovně. GTS-Root R1 je certifikační autoritou střední úrovně. R1 je nejvyšší kořenová certifikační autorita společnosti GlobalSign. Touto cestou lze sestavit řetězec důvěry.



Obr. 35. Příklad řetězce důvěry

Řetězec důvěry má 3 části:

- **Kořenový certifikát** je digitální certifikát, který je vlastnictvím certifikační autority, která jej vydala. Většina prohlížečů jej má například předinstalovaný a je uložen v „*důvěryhodném úložišti*“. Certifikační autority na kořenové certifikáty bedlivě dohlížejí. Například kořenová certifikační autorita GlobalSign Root CA – R1 je kořenovou certifikační autoritou.
- **Zprostředkující certifikáty** jsou větvemi pomyslného stromu a kořenový certifikát je tedy jeho kmenem. Slouží jako spojovací článek mezi chráněnými kořenovými certifikáty a veřejně vydanými certifikáty serveru. V řetězci bude vždy alespoň jeden zprostředkující certifikát, ale může jich být samozřejmě i více.
- **Certifikát serveru** je takový certifikát, který byl přidělen určité doméně (v tomto případě např. www.google.com).

7.4 Ověřování pomocí PKI a PKC

V digitálním světě je infrastruktura veřejných klíčů (PKI) systémem pro ověřování osob a zařízení. Jedna nebo více důvěryhodných stran digitálně podepisuje dokumenty ověřující, že konkrétní kryptografický klíč patří konkrétnímu uživateli nebo zařízení. Klíč pak může být použit jako identita uživatele v digitálních sítích.

Digitální certifikáty lze použít také pro 2FA nebo pro ověřování bez hesla.

Při pokusu o ověření identity uživatele na serveru vytvoří server náhodná data a odešle je uživateli. Uživatel pak data zašifruje pomocí svého soukromého klíče a vrátí je zpět serveru. Server data dešifruje pomocí veřejného klíče digitálního certifikátu uživatele, a pokud se dešifrovaná data shodují s přijatými daty, server ví, že uživatel je tím, za koho se vydává. Toto je základní proces použití PKI+PKC pro účely ověřování.

KAPITOLA 8

Test

Nejbezpečnější způsob ukládání hesel je:

- ◯ šifrování
- ◯ hashování
- ◯ otevřený text
- ◯ solené hashování

Co je smyslem autentifikace?

- ☐ ověřit identitu
- ☐ identifikace
- ☐ zkontrolovat, k jakým prostředům má dotyčný přístup
- ☐ je podstatnou částí kyberbezpečnosti

Solení hesel znesnadňuje útočníkovi útok, protože slovníkový útok je specifický pro:

- ◯ každého uživatele
- ◯ každého útočníka
- ◯ každé zařízení
- ◯ každé heslo

Jaké jsou nejběžnější metody ověřování?

- ☐ uživatelské jméno a heslo
- ☐ sken tváře

- ☐ e-mail a heslo
- ☐ RSA Secure ID

Jaká je největší nevýhoda autentizace založené na vlastnosti "podle toho, čím uživatel je"?

- ◯ je neodvolatelná
- ◯ může být ztracená
- ◯ může být zapomenutá
- ◯ může být odcizená

Jaké jsou hlavní kategorie ověřování?

- ☐ jak vypadáte
- ☐ podle toho, co znáte
- ☐ podle toho, kým jste
- ☐ podle toho, co máte

Co je to vícefázová autentifikace?

- ◯ Autentifikace, která využívá alespoň dva různé identifikátory pre ověření identity.
- ◯ Autentifikace, která využívá přesně dva různé identifikátory pre ověření identity.
- ◯ Autentifikace, která využívá přesně jeden identifikátor pre ověření identity.
- ◯ Je stejná jako dvoufázová autentifikace (2FA).

K ověření metodou „podle toho, co máte“ je možné využít:

- ☐ chytrý telefon
- ☐ heslo
- ☐ otisk prstu
- ☐ čipová karta

Komu se u metody autentifikace typu výzva-odpověd posílá výzva?

- ◯ uživateli
- ◯ serveru
- ◯ programu
- ◯ ověřovateli

Základní proces autentifikace zahrnuje:

- ☐ server
- ☐ autentifikační údaje
- ☐ uživatel
- ☐ kľíčové slovo

Součástmi ověřovacího mechanismu jsou:

- ☐ vstupní zařízení
- ☐ ověřovatel
- ☐ počítač
- ☐ distribuční systém

Jaká je doporučená délka hesla podle aktuálně platných pokynů a osvědčených postupů?

- ◯ 6 znaků
- ◯ 4 znaky
- ◯ 7 znaků
- ◯ alespoň 12 znaků

K čemu slouží hashovací funkce?

- ◯ vytvářejí hashtag
- ◯ vypočítávají jedinečný identifikátor údajů
- ◯ vytvářejí hesla
- ◯ chrání proces autentifikace

Které prvky (faktory) je možné využít u dvojfázové autentifikace?

- ☐ zpráva SMS
- ☐ identifikační token
- ☐ aplikace v chytrém telefonu
- ☐ uživatelské jméno

Která z následujících odpovědí je standardem pro autentifikaci bez hesla?

- ◯ FIDO2
- ◯ FIBA
- ◯ UFI
- ◯ UPA

Současné průmyslové standardy pro bezpečné ukládání hesel zahrnují:

- ☐ šifrování
- ☐ hashování
- ☐ ukládání v otevřeném formátu
- ☐ solené hashování

V jakém pořadí se u procesu digitálního podepisování používají klíče?

- ◯ soukromý klíč odesílatele a veřejný klíč příjemce
- ◯ soukromý klíč odesílatele a veřejný klíč odesílatele

- ◯ soukromý klíč příjemce a veřejný klíč příjemce
- ◯ veřejný klíč odesílatele a soukromý klíč příjemce

Jaký druh útoků znesnadňuje technika solení hesel?

- ☐ slovníkové útoky
- ☐ útoky hrubou silou
- ☐ útoky na server
- ☐ útoky na mobilní zařízení

Jaké jsou slabiny technik ověřování v kategorii "podle toho, co znáte"?

- ☐ faktor ověření může být zapomenutý
- ☐ faktor ověření může být ztracen
- ☐ faktor ověření může být okopírován
- ☐ faktor ověření může být zneplatněn

Pokud se pro účely šifrování používá kryptografie s veřejným klíčem, jaký klíč se použije k zašifrování dat?

- ◯ veřejný klíč příjemce
- ◯ veřejný klíč odesílatele
- ◯ soukromý klíč příjemce
- ◯ soukromý klíč odesílatele

Jaké jsou slabiny technik ověřování v kategorii "podle toho, co máte"?

- ☐ faktor ověření může být zapomenutý
- ☐ faktor ověření může být ztracen
- ☐ faktor ověření může být okopírován

☐ faktor ověření může být zneplatněn

Jaká posloupnost klíčů se používá při asymetrickém šifrování?

◯ soukromý klíč odesílatele a veřejný klíč příjemce

◯ soukromý klíč odesílatele a veřejný klíč odesílatele

◯ soukromý klíč příjemce a veřejný klíč příjemce

◯ veřejný klíč odesílatele a soukromý klíč příjemce

Jaká je správná posloupnost kroků v procesu digitálního podepisování?

◯ hashování, podpis a odeslání

◯ podpis, hashování a odeslání

◯ šifrování, hashování a odeslání

◯ hashování, kódování a odeslání

Jaké jsou příklady ověřování heslem?

☐ jednorázová hesla

☐ opětovně použitelná hesla

☐ strukturovaná hesla

☐ osobní údaje

Jaké jsou komponenty infrastruktury PKI (infrastruktura veřejných klíčů)?

◯ CA, MA, LA, digitální podpis

◯ CA, RA, PA, digitální podpis

◯ CA, RA, PKC, digitální certifikát

◯ CA, RA, PKC, digitální podpis

Vyberte z následujících možností neelektronické útoky na hesla:

- ☐ sledování potenciální oběti
- ☐ sociální inženýrství
- ☐ slovníkový útok
- ☐ útok hrubou silou

Jaký je hlavní úkol certifikační autority?

- ◯ vydávání certifikátů
- ◯ vydávání digitálních podpisů
- ◯ kontrola digitálních podpisů
- ◯ ověřování identity jednotlivců

Vyberte z následujících možností elektronické útoky na hesla:

- ☐ phishing
- ☐ sociální inženýrství
- ☐ slovníkový útok
- ☐ útok hrubou silou

Jakým způsobm je implementováno PKI?

- ◯ formou stromové struktury
- ◯ jako systém klient-server
- ◯ hardwarově
- ◯ sekvenčně

Které z následujících možností jsou nástroje pro prolamování hesel?

- ☐ John the cracker
- ☐ John the ripper
- ☐ Hydra
- ☐ Hybrid

Co by hesla neměla v žádném případě obsahovat?

- ☐ odlišné typy znaků
- ☐ slova spojená s vaší osobou
- ☐ data narození
- ☐ speciální znaky

Jaké funkce obvykle vykonává správce hesel?

- ☐ automatické doplňování
- ☐ generování hesel
- ☐ hodnocení kvality hesel
- ☐ zneplatnění hesel

Jaké nevýhody má autentifikace typu 2FA?

- ☐ nepohodlí
- ☐ vyšší bezpečnost
- ☐ obavy ze ztráty soukromí
- ☐ robustnější autentifikace

Co je hlavním smyslem digitálních podpisů?

- ☐ autentifikace
- ☐ integrita dat

- ☐ důvěryhodnost dat
- ☐ autorizace

Jaký typy klíčů se používají při asymetrickém šifrování?

- ☐ veřejný klíč
- ☐ tajný klíč
- ☐ přihlašovací klíč
- ☐ soukromý klíč

Jaké stavební prvky jsou součástí procesu digitálního podepisování?

- ☐ hashovací funkce
- ☐ symetrické šifrovací algoritmy
- ☐ algoritmy na výměnu klíčů
- ☐ asymetrické šifrovací algoritmy

Jaká je správná posloupnost kroků v procesu ověřování digitálního podpisu?

- ☐ získat hash z podpisu, získat hash z údajů, porovnat je navzájem
- ☐ získat hash z údajů, získat hash z podpisu, porovnat je navzájem
- ☐ porovnání hashů, následně získání hashe z údajů a získání hashe z podpisu
- ☐ porovnání hashů, následně získání hashe z podpisu a potom získání hashe z údajů

Které součásti jsou součástí infrastruktury PKI (infrastruktury veřejných klíčů)?

- ☐ certifikační autorita (CA)
- ☐ provozní autorita (PA)
- ☐ digitální podpis
- ☐ digitální certifikát

Jaké jsou typické součásti digitálních certifikátů?

- ☐ doba platnosti
- ☐ vydavatel
- ☐ velikost
- ☐ digitální podpis

Jaké jsou hlavní části důvěryhodného řetězce PKI?

- ☐ kořenový certifikát
- ☐ zprostředkovací certifikát
- ☐ digitální podpis
- ☐ administrativní certifikát

Jakým způsobem lze využít digitální certifikáty k ověřování?

- ☐ slouží jako hlavní autentifikační faktor
- ☐ slouží jako druhý autentifikační faktor
- ☐ nedají se použít
- ☐ na podepisování dokumentů