Bezpečnostné zlyhania a ich príčiny (niektoré príklady z praxe)

Martin Stanek

Katedra informatiky
FMFI UK, Bratislava
stanek@dcs.fmph.uniba.sk

2012/13

Obsah

Bezpečnostné zlyhania a zraniteľnosti
- všeobecne

Konkrétne bezpečnostné zlyhania a zraniteľnosti
- Náhodnosť kryptografických kľúčov
- Timing útoky na porovnávanie
- DLL Hijacking
- PKCS #11 a kryptografické tokeny
- WPS (WiFi Protected Setup)
- Šifrované USB kľúče
- Kolízie v hašovacích tabuľkách

Medializované incidenty

Úvod

Bezpečnostné zlyhania

- môžu mať rôzne príčiny (alebo ich kombináciu): ľudský faktor, kriminálna aktivita, technické nedostatky (zraniteľnosti) a pod.
- majú rôzne dopady: „nič sa nestalo“, strata reputácie, náklady na obnovu/náhradu systémov, bankrot a pod.

V tejto prednáške sa venujeme najmä zlyhaniam technického charakteru (zraniteľnostiam).

- nie je to vyčerpávajúci zoznam . . . realita je horšia
- National Vulnerability Database (nvd.nist.gov)

Typy zraniteľností podľa NVD (23)

- Authentication Issues
- Credentials Management
- Permissions, Privileges, and Access Control
- Buffer Errors
- Cross-Site Request Forgery (CSRF)
- Cross-Site Scripting (XSS)
- Cryptographic Issues
- Path Traversal
- Code Injection
- Format String Vulnerability
- Configuration
- Information Leak / Disclosure
- Input Validation
- Numeric Errors
- OS Command Injections
- Race Conditions
- Resource Management Errors
- SQL Injection
- Link Following
- Other
- Not in CWE
- Insufficient Information
- Design Error

NVD – zraniteľnosti publikované v roku 2012

- Authentication Issues (1,87%)
- Credentials Mngt. (0,98%)
- Permissions, . . . (11,42%)
- Buffer Errors (13,69%)
- CSRF (2,89%)
- XSS (13,61%)
- Cryptographic Issues (1,74%)
- Path Traversal (2,23%)
- Code Injection (2,59%)
- Format String Vuln. (0,21%)
- Configuration (0,59%)
- Information Leak (4,10%)
- Input Validation (7,00%)
- Numeric Errors (2,87%)
- OS Command Inj. (0,26%)
- Race Conditions (1,30%)
- Resource Mngt. Errors (6,01%)
- SQL Injection (4,50%)
- Link Following (0,32%)
- Other (4,97%)
- Not in CWE (0,04%)
- Insufficient Infor. (15,41%)
- Design Error (1,68%)

celkovo 5289 zraniteľností

Iné klasifikácie zraniteľností

- Common Weaknesses Enumeration (cwe.mitre.org)
  - Common Attack Pattern Enumeration and Classification (capec.mitre.org)
- Open Web Application Security Project (OWASP, www.owasp.org)
  - primárne web aplikácie – zraniteľnosti, útoky, riziká
  - príručka pre testovanie
- podrobnejšia klasifikácia, popis, (niekde) príklady a dodatočné informácie

Reálny svet

- nie každá zraniteľnosť má reálny a rovnako veľký dopad
- najčastejšie útoky s dôsledkom úniku dát: SQL Injection a XSS

  zdroj: IBM X-Force 2012 Mid-Year Trend and Risk Report

- najčastejšie príčiny úniku dát v 2012:
  1. Hack (57%)
  2. Podvod a sociálne inžinierstvo (9%)
  3. Web (dáta k dispozícii cez vyhľadávače a pod.) (5%)
  4. Ukradnutý notebook (5%)
  5. Neznáma príčina (4%)
  6. Nesprávna likvidácia dokumentov (4%)

     ...

     zdroj: DataLossDB (datalossdb.org)

Top hrozby a zraniteľnosti v reálnom svete

1. Nedbanliví zamestnanci
2. Útoky s cieľom krádeže finančných informácií
3. Zastarané bezpečnostné opatrenia
4. Útoky s cieľom poškodiť organizáciu
5. Podvod
6. Prírodná katastrofa

   ...

zdroj: Ernst & Young's 2012 Global Information Security Survey, na základe globálneho prieskumu (64 krajín, 1836 respondentov)

Top priority v reálnom svete

1. Kontinuita činnosti/havarijné plánovanie
2. Tranformácia (fundamentálna zmena) inf. bezpečnosti
3. Prevencia únikov dát (technológie a procesy)
4. Implementácia štandardov (napr. ISO/IEC 27002)
5. Riadenie rizík
6. Zabezpečenie nových technológií (cloud, mobilné technológie a pod.)

   ...

zdroj: Ernst & Young's 2012 Global Information Security Survey

Niekoľko zraniteľností . . .

ilustračné príklady

Náhodnosť kryptografických kľúčov

- 2008 – Debian
- vlastná úprava openssl (použitie neinicializovanej pamäte)
  - úprava na zlom mieste
  - poškodená inicializácia pseudonáhodného generátora . . . inicializovaný iba PID bežiaceho procesu
  - zohľadniac konkrétnu architektúru dokopy len 98301 rôznych inicializačných hodnôt
- dopady:
  - predikovateľné kľúče pre SSH, OpenVPN, DNSSEC, X.509 certifikáty, session kľúče v SSL/TLS spojeniach, . . .
  - použitie knižnice na 1 podpis DSA . . . kompromitovaný súkromný kľúč

Neskôr . . . v openssl 1.0

```
/* DO NOT REMOVE THE FOLLOWING CALL TO MD_Update()! */
MD_Update(&m,buf,j);
/* We know that line may cause programs such as
   purify and valgrind to complain about use of
   uninitialized data.  The problem is not, it's
   with the caller.  Removing that line will make
   sure you get really bad randomness and thereby
   other problems such as very insecure keys. */
```

- Dobre implementovať kryptografiu nie je ľahké
  - 6 zraniteľností od roku 2010 s vysokou závažnosťou v openssl

Timing útoky na porovnávanie (Google, Sun, . . .)

- 2009 – Keyczar (Google), Java (Sun), . . .
- situácia: server porovnáva poslaný HMAC s korektným (ktorý si vypočíta sám)
- potenciálny cieľ útočníka: získať korektný HMAC k vlastnej správe (ktorá sa následne javí „autenticky“)
- čo je zlé na nasledujúcom kóde (Python)?

```
return self.Sign(msg) == sig_bytes
```

Čo je zlé na nasledujúcom kóde (Java)?

```
public static boolean
isEqual(byte digesta[], byte digestb[]) {
  if (digesta.length != digestb.length)
    return false;

  for (int i = 0; i < digesta.length; i++) {
    if (digesta[i] != digestb[i])
      return false;
  }
  return true;
}
```

Rekonštrukcia HMAC

Ako dlho trvá serveru odpoveď o nesprávnom HMAC:

- ak správne uhádneme prvých 0, 1, 8 alebo 15 bajtov hodnoty?
- rekonštrukcia HMAC na základe časovej variability odpovedí
- pre štvrtý bajt:

```
71 A0 89 00 00 ... 00
71 A0 89 01 00 ... 00
         ...
71 A0 89 4A 00 ... 00     dlhší čas spracovania?
         ...
71 A0 89 FF 00 ... 00
```

- vyžaduje viacero meraní pre konkrétnu hodnotu (šum)
- štatistické vyhodnotenie meraní

Časovo konštantné porovnanie (Java)

```
public static boolean
isEqual(byte[] digesta, byte[] digestb) {
  if (digesta.length != digestb.length)
    return false;

  int result = 0;
  for (int i = 0; i < digesta.length; i++) {
    result |= digesta[i] ^ digestb[i];
  }
  return result == 0;
}
```

DLL Hijacking

- 2010, 2011 – Microsoft a ďalší tvorcovia aplikácií
- nahratie a vykonanie falošnej (podstrčenej) DLL legitímnou aplikáciou
- ako a odkiaľ nahráva aplikácia DLL?
  - bez špecifikácie úplnej cesty k DLL
  - z aktuálneho pracovného adresára (potenciálne sieťový zdieľaný disk, USB disk a pod.)
  - prípadné ďalšie situácie

PKCS #11 a kryptografické tokeny

- PKCS #11 – Cryptographic Token Interface Standard
  - definuje rozhranie pre komunikáciu a prácu s tokenmi
  - tokeny: HSM, čipové karty (v rôznych formách), softvérové tokeny
  - generovanie kľúčov, šifrovanie a dešifrovanie, podpisovanie a overovanie podpisov, ...
- súkromné a tajné kľúče môžu mať atribúty
  - *sensitive* – nemôžu byť získané z tokenu v otvorenom tvare
  - *unextractable* – nemôžu byť získané z tokenu ani zašifrované
  - *wrap* – kľúčom možno šifrovať iný kľúč
  - *decrypt* – kľúčom možno dešifrovať dáta
  - ... niektoré ďalšie pravidlá o nastavovaní a kombinovaní atribútov
- štandard nie je vždy explicitný; výrobcovia implementujú rôzne

Slabiny implementácií PKCS #11

- niektoré tokeny umožňujú export citlivých alebo neextrahovateľných kľúčov v otovrenom tvare (v rozpore so štandardom)
- wrap/decrypt útok:
  - kľúče k_1 (sensitive), k_2 (wrap, decrypt)
    1. wrap ... $\{k_1\}_{k_2}$
    2. decrypt ... k_1
  - variant s vložením nového kľúča k_2 do tokenu (následne stačí len wrap)
  - variant s vytvorením kópie k_2, pričom jedna bude „wrap“ a druhá „decrypt“
- iné problémy súvisiace s možnosťou meniť atribúty kľúčov

WPS (WiFi Protected Setup)

- december 2011
- cieľ: uľahčiť používateľom pripájanie zariadení do WiFi siete
- realizácia:
  - autentizácia pomocou 8 ciferného PIN kódu (uvedeného na nálepke na zariadení)
  - teoreticky 10^8 možností
  - prakticky: z odpovede pri nesprávnom PIN je možné určiť, či je prvá polovica PIN zadaná správne; posledná cifra PIN je kontrolná cifra . . . teda $10^4 + 10^3$ možností
- niektoré smerovače neumožňujú WPS vypnúť

Šifrované USB kľúče

- január 2010; Kingstone, SanDisk, Verbatim
- certifikácia podľa FIPS 140-2 Level 2, šifrovanie: AES-256
- realita:
  - šifrovací kľúč nezávisí na hesle
  - USB kľúč sa „odomkne“, ak dostane očakávaný reťazec (konštantný, bez ohľadu na heslo)

Kolízie v hašovacích tabuľkách

- december 2011; Oracle, Microsoft, PHP, Apache Tomcat, ...
- analogický problém pôvodne 2003; Perl, Squid
- hašovacie tabuľky – dátové štruktúry pre ukladanie/vyberanie dát podľa nejakého kľúča
  - priemerný prípad: $O(n)$ vloženie/vymazanie/nájdenie n prvkov
  - najhorší prípad: $O(n^2)$ pre n prvkov (kolízie)
- problém: vieme nájsť veľa kolízií rôznych kľúčov
- automatické parsovanie parametrov POST dát v rámci HTTP do hašovacích tabuliek
- DoS útok na web server

„Normálne“ hašovanie

- Java 6 (java.lang.String, metóda: public int hashCode())
  - 32-bitová aritmetika (int), s_i je i-ty znak reťazca ($s_{1,\ldots,n}$):

$$\sum_{i=1}^{n} 31^{n-i} \cdot s_i$$

- PHP 5 (algoritmus DJBX33A, 32 bitová aritmetika), s_0 je konštanta 5381

$$\sum_{i=0}^{n} 33^{n-i} \cdot s_i$$

- ASP.NET (algoritmus DJBX33X), s_0 je konštanta 5381

$$\bigoplus_{i=0}^{n} 33^{n-i} \cdot s_i$$

- ľahko hľadať veľký počet multikolízií

Riešenia

- dočasné – obmedzenie veľkosti POST dotazov, limit CPU pre jeden dotaz a pod.
- trvalé – lepšia hašovacia funkcia
  - napr. znáhodnené hašovanie – funkcia oplyvnená náhodným parametrom pri štarte procesu

Medializované incidenty (1)

- Sony
  - apríl, máj 2011; krádež osobných údajov cca. 77 miliónov používateľov (PlayStation Network, Qriocity)
  - Sony (PSN blog): *The "illegal and unauthorized person" obtained people's names, addresses, email address, birth dates, usernames, passwords, logins, security questions and more.*
- Stuxnet
  - cielený útok na priemyselné systémy
  - červ špecificky určený pre PC so SCADA softvérom Siemensu
  - objavený: jún/júl 2010; používajúci 4 zero-day zraniteľnosti Windows
  - sofistikovaný, veľký, drivre podpísané ukradnutými platnými kľúčmi (platné certifikáty verejných kľúčov)
  - súvisiaca podobná infiltrácia: Duqu (zber informácií)

Medializované incidenty (2)

- Aurora
  - druhá polovica 2009, cielený útok na Google, Adobe, Juniper, ...
  - využitá zraniteľnosť IE (zero-day zraniteľnosť)
  - primárny záujem (podľa McAfee) – prístup k zdrojovým kódom a pod.
- RSA
  - marec 2011; cielený útok na RSA (divízia EMC)
  - SecurID tokeny (jednorazové heslá, dvojfaktorová autentizácia), väčšinový podiel na trhu
  - potrebné nahradiť 40 miliónov tokenov

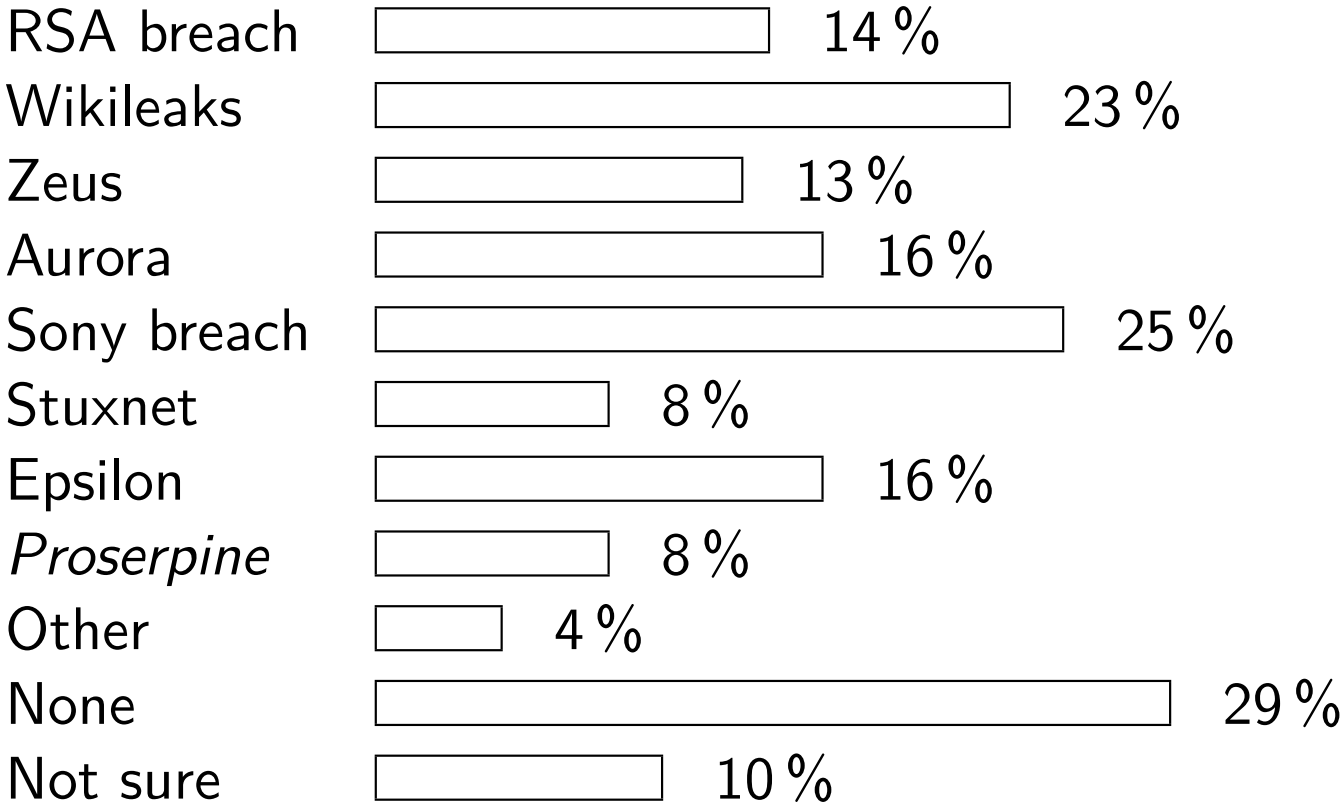
Medializované incidenty (3)

- Zeus
  - od 2007, rozšírenie 2009, 2010
  - trójsky kôň, viacero botnetov (niekoľko miliónov kompromitovaných počítačov)
  - krádeže prihlasovacích údajov – sociálne siete, e-mailové účty, bankové účty, ...
- Epsilon
  - najväčší svetový poskytovateľ e-mail marketingu
  - apríl 2011; ukradnuté mená a e-mailové adresy
  - JP Morgan Chase, Citibank, American Express, ...
- ďalšie incidenty únikov dát:
  - `http://datalossdb.org/`, `http://www.databreaches.net`, ...

Medializované incidenty (4) – certifikačné autority

- Comodo
  - marec 2011; 9 falošných certifikátov
  - kompromitácia používateľského účtu v registračnej autorite
- DigiNotar
  - holandská CA (kompromitovaná od 2009),
  - júl 2011 – falošný „wildcard“ certifikát pre `*.google.com` (celkovo niekoľko stoviek falošných certifikátov)
  - september 2011 – bankrot

Vplyv medializovaných incidentov na prax



na základe odpovedí 1000 IT manažérov UK, USA, Kanade a Austrálii

Zdroj: Websense Research Report: *Security Pros & "Cons"*, 2011