Security incidents, weaknesses and vulnerabilities

Martin Stanek

Department of Computer Science
Comenius University
stanek@dcs.fmph.uniba.sk

Security of IT infrastructure (2015/16)

Content

Introduction

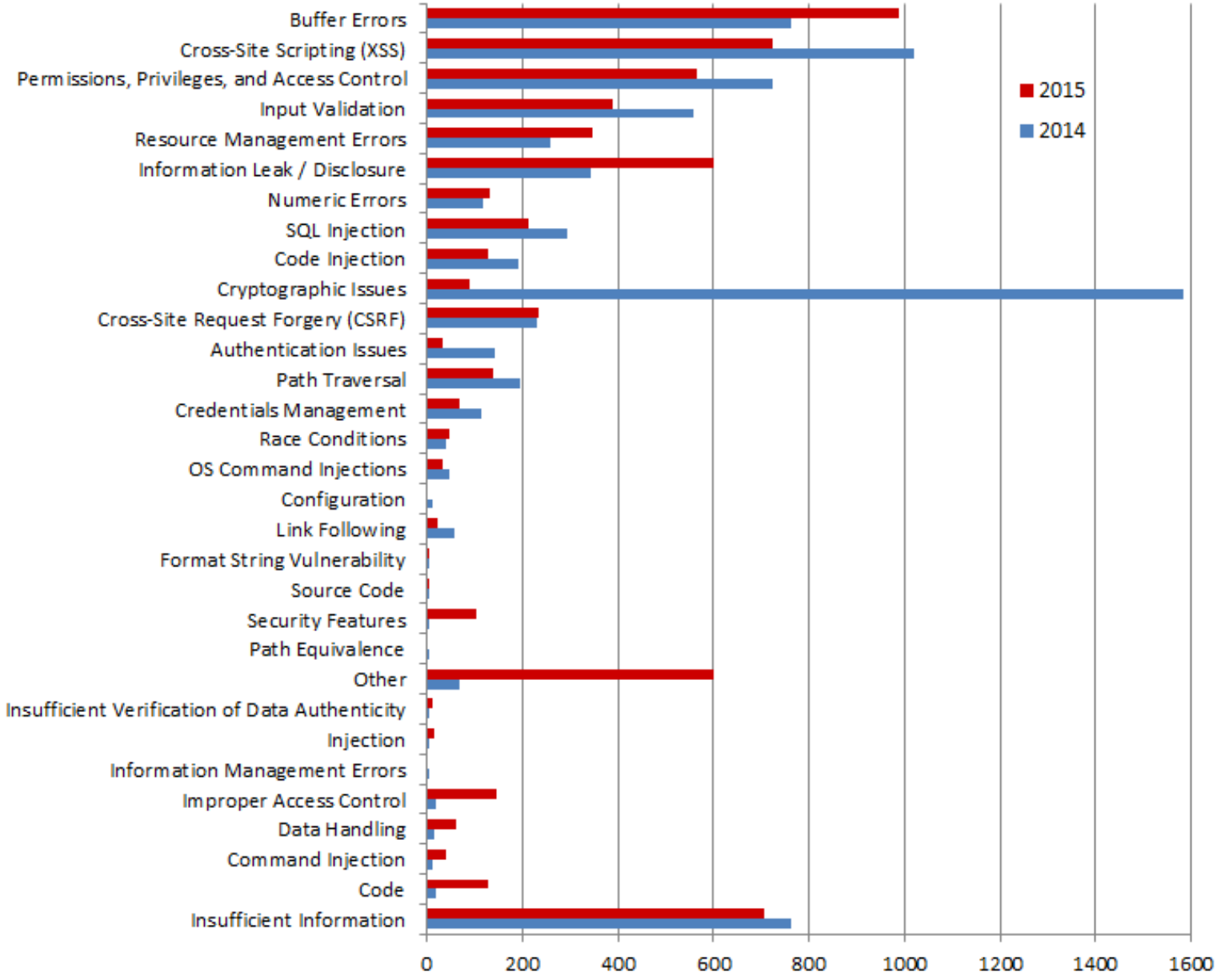
Security incidents and failures

- various causes (or their combination): human factor, criminal activities, technical vulnerabilities etc.
- impact: “nothing” happened, loss of reputation, cost of repair/replacement of data and systems, direct financial loss, bankruptcy etc.

mostly technical failures/vulnerabilities in this lecture

- just examples . . . reality is worse (unpublished vulnerabilities, weak passwords, misconfiguration, etc.)
- National Vulnerability Database (nvd.nist.gov)
- vulnerabilities (software flaws) published:
  5278 (2012), 5174 (2013), 7903 (2014), 6453 (2015)
- classification (categories, severity etc.)

NVD – vulnerabilities published in 2014 and 2015



Strange number of “Cryptographic Issues” in 2014

- January – August: 125 vulnerabilities
- September: 674 vulnerabilities
- October: 712 vulnerabilities
- November – December: 22 vulnerabilities

- *The . . . application for Android does not verify X.509 certificates from SSL servers, which allows man-in-the-middle attackers to spoof servers and obtain sensitive information via a crafted certificate.*
- (?) nogotofail tool released by Google in November 2014

Examples . . . (1)

Authentication Issues (CVE-2015-7755):
Juniper ScreenOS 6.2.0r15 through 6.2.0r18, . . . , and 6.3.0r20 before 6.3.0r21 allows remote attackers to obtain administrative access by entering an unspecified password during a (1) SSH or (2) TELNET session.
user: auth_admin_internal, password: `<<< %s(un='%s') = %u`

Buffer Errors (CVE-2016-0946):
Adobe Reader and Acrobat before 11.0.14, Acrobat and Acrobat Reader DC Classic before 15.006.30119, and ... on Windows and OS X allow attackers to execute arbitrary code or cause a denial of service (memory corruption) via unspecified vectors, a different vulnerability than CVE-2016-0931, CVE-2016-0933, CVE-2016-0936, CVE-2016-0938, CVE-2016-0939, CVE-2016-0942, CVE-2016-0944, and CVE-2016-0945.

Examples . . . (2)

Credentials Management (CVE-2015-7283):
The web administration interface on ZyXEL NBG-418N devices with firmware 1.00(AADZ.3)C0 has a default password of 1234 for the admin account, which allows remote attackers to obtain administrative privileges by leveraging a LAN session.

Cryptographic Issues (CVE-2015-2233):
Lenovo System Update (formerly ThinkVantage System Update) before 5.06.0034 does not properly validate CA chains during signature validation, which allows man-in-the-middle attackers to upload and execute arbitrary files via a crafted certificate.

Improper Access Control (CVE-2015-3306):
The mod_copy module in ProFTPD 1.3.5 allows remote attackers to read and write to arbitrary files via the site cpfr and site cpto commands. (without authentication)

Examples . . . (3)

SQL Injection (CVE-2015-5308):
Multiple SQL injection vulnerabilities in cs_admin_users.php in the wp-championship plugin 5.8 for WordPress allow remote attackers to execute arbitrary SQL commands via the (1) user, (2) isadmin, (3) mail service, (4) mailresceipt, (5) stellv, (6) champtipp, (7) tippgroup, or (8) userid parameter.

Security Features (CVE-2016-0019):
The Remote Desktop Protocol (RDP) service implementation in Microsoft Windows 10 Gold and 1511 allows remote attackers to bypass intended access restrictions and establish sessions for blank-password accounts via a modified RDP client, aka “Windows Remote Desktop Protocol Security Bypass Vulnerability.”

Other classifications of vulnerabilities

- MITRE:
  - Common Vulnerabilities and Exposures (cve.mitre.org)
  - Common Weaknesses Enumeration (cwe.mitre.org)
  - Common Attack Pattern Enumeration and Classification (capec.mitre.org)
- Open Web Application Security Project (OWASP, www.owasp.org)
  - primarily for web applications – vulnerabilities, attacks, risks
  - OWAPSP Top 10 (most critical web application security risks, 2013)
  - Testing Guide (v4, 2014)
- more detailed classifications, description, examples, additional information

Real world – surveys, analyses, predictions

- EY's Global Information Security Survey 2015 (October 2015)
- Various Security Predictions for 2016:
  - Symantec, Raytheon/Websense, McAfee (Intel Security), FireEye, Trend Micro, IBM, Kaspersky, ...
- DataLossDB Statistics (datalossdb.org)
- Verizon 2015 Data Breach Investigations Report
- ...etc.

EY's Global Information Security Survey 2015

- 1755 respondents, from 67 countries
- 88% of respondents do not believe their information security fully meets the organization's needs
- top two threats: phishing, malware
- 36% of organizations are unlikely to detect a sophisticated cyber attack
- the most likely source of attack: criminal syndicates, employee, hacktivists, ...

Verizon 2015 Data Breach Investigations Report

- summary of 2014
- 79790 security incidents, 2122 data breaches, 61 countries represented
- “99.9% of the exploited vulnerabilities had been compromised more than a year after the associated CVE was published”

  Top vulnerabilities:
  - CVE-2002-0012, CVE-2002-0013 (target: SNMP implementations)
  - CVE-1999-0517: An SNMP community name is the default (e.g. public), null, or missing.
  - CVE-2001-0540 (target: terminal servers Windows NT/2000)
  - CVE-2014-3566 (target: OpenSSL – POODLE attack)
  - CVE-2014-0152 (target: RDP service in Windows Server 2008 R2 and R2 SP1 and Windows 7 Gold and SP1)

Trend Micro Security Predictions 2016

- 2016 will be the Year of Online Extortion
- At least one consumer-grade smart device failure will be lethal in 2016.
- China will drive mobile malware growth to 20M by the end of 2016; globally, mobile payment methods will be attacked.
- Data breaches will be used by hacktivists to systematically destroy their targets.
- Despite the need for Data Protection Officers, less than 50% of organizations will have them by end of 2016.
- Ad-blocking will shake up the advertising business model and kill malvertisements.
- Cybercrime legislation will take a significant step towards becoming a truly global movement.

The most frequent types of data breaches 2015 (DataLossDB.org)

1. Hack – Computer-based intrusion (34%)
2. Skimming (15%)
3. Fraud or scam (usually insider-related), social engineering (8%)
4. Other – Breach type was disclosed but no formal classification (5%)
5. Discovery of documents not disposed properly (4%)
6. Virus (4%)
7. Web – Data typically available via search engines, public pages, etc. (4%)
8. Unknown or unreported breach type (3%)
9. Email communication exposed to unintended third party (3%)
10. Stolen Laptop (generally specified as a laptop in media reports) (3%)
11. Snooping – Employee exceeding intended privileges (4%)
12. SnailMail (4%)
13. Documents either reported or known to have been stolen by a third party (3%)
14. Phishing (2%)

... other causes

Data breaches – examples (1)

1. University of Veterinary Medicine and Pharmacy in Košice
   - January 2014
   - leaked personal data of 1500 students
   - addresses, ID card numbers, personal identification numbers, ...
   - published PDF in Central register of contracts
   - "blacked out" personal data ...still there, can be copied (selected)
2. Anthem (managed health care company)
   - December 2014 – January 2015
   - leaked personal data of 80 million customers
   - names, dates of birth, SSN, health care ID numbers, home addresses, email addresses, employment information, income data
   - attack: some tech employees had their credentials compromised
   - detection: noticing suspicious queries

   Similar breach: Premera (11 million people)

Data breaches – examples (2)

3. Ashley-Madison
   - data breach announced in July 2015 (“Impact Team”)
   - 10GB + 19GB compressed data
   - $\sim$ 37 million customer records
   - e-mail addresses, names, credit card transactions, …
   - source code, e-mails
   - suicides, blackmailing, bcrypt + MD5
4. Office of Personnel Management
   - detected: April 2015, started: March 2014
   - 21.5 million records
   - attackers with valid user credentials / contractors
   - names, SSNs, dates and places of birth, addresses, security-clearance information
   - 5.6 million sets of fingerprints

Data breaches – examples (3)

5. Hacking Team
   - selling offensive intrusion and surveillance capabilities to governments, law enforcement agencies and corporations
   - data breach announced: July 2015
   - 400GB (customers, e-mails, 0-day exploits, source code, ...)
   - weak passwords, e.g. “P4ssword”, “HTPassw0rd”, “wolverine”
6. IRS (Internal Revenue Service)
   - detected: too many old tax returns (May 2015)
   - stolen credentials (probably from other data breaches, e.g. Anthem)
   - 334 thousand people
   - 15,000 falsified documents processed ... 50 million USD in refunds

Data breaches – examples (2)

7. Target (USA, retail)
   - December 2013
   - 40 million credit and debit cards information
     + additional 70 million personal information
   - card information
     + names, mailing addresses, phone numbers, email addresses
   - malware installed on POS devices
   - entry using authentication credentials stolen from a heating, ventilation, and air-conditioning subcontractor
8. JPMorgan Chase (USA, banking)
   - discovered in July 2014
   - names, addresses, phone numbers and e-mail addresses of 83 million account holders
   - initial assumption: 0-day web server exploit (?)
   - reality: stolen credentials (password), 2nd factor not enabled on one server
   - 90 servers compromised when detected

Data breaches – examples (3)

9. Home Depot (USA, retail)
   - breach started in April 2014, undetected for 5 months
   - 56 million customer credit and debit card accounts
     + 53 million customer email addresses
   - malware on self-checkout registers
   - initial step: credentials stolen from a third-party vendor
10. Sony Pictures (USA, entertainment)
   - 100TB of data (?)
   - discovered in November 2014
   - personal information about employees, e-mails, salaries, copies of unreleased Sony films
   - North Korea vs. current/former employees (?)
   - the White House reacts

. . . and finally:

- stealing money
  - Kaspersky Lab published their findings (February 2015)
  - Carbanak (malware)
  - more than 100 banks in 30 countries (Russia, Japan, USA, . . .)
  - more than 300 million USD stolen
  - initial vector: spear phishing attacks
  - similar attacks continue till today (Metel, GCMan, Carbank 2.0)
  - another findings published in February 2016
- Ukrainian Power Grid
  - December 2015
  - BlackEnergy trojan
  - black-out (for few hours): 103 cities complete 184 cities partial
  - blocked call centers

SANS Critical Security Controls (1)

https://www.sans.org/critical-security-controls

1. Inventory of Authorized and Unauthorized Devices
2. Inventory of Authorized and Unauthorized Software
3. Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers
4. Continuous Vulnerability Assessment and Remediation
5. Controlled Use of Administrative Privileges
6. Maintenance, Monitoring, and Analysis of Audit Logs
7. Email and Web Browser Protections
8. Malware Defenses
9. Limitation and Control of Network Ports, Protocols, and Services
10. Data Recovery Capability

SANS Critical Security Controls (2)

https://www.sans.org/critical-security-controls

11. Secure Configurations for Network Devices such as Firewalls, Routers, and Switches
12. Boundary Defense
13. Data Protection
14. Controlled Access Based on the Need to Know
15. Wireless Access Control
16. Account Monitoring and Control
17. Security Skills Assessment and Appropriate Training to Fill Gaps
18. Application Software Security
19. Incident Response and Management
20. Penetration Tests and Red Team Exercises

Security failures/vulnerabilities ...

examples

Randomness of cryptographic keys

- 2008 – Debian
- modification of openssl source code
  - the use of uninitialized memory
  - broken initialization of pseudorandom generator . . . initialized by PID only
  - at most 98301 unique initialization values overall (depending on particular platform)
- impact:
  - predictable keys for SSH, OpenVPN, DNSSEC, X.509 certificates, session keys in SSL/TLS, . . .
  - using library just for a single DSA signing . . . compromised private key
  - similar problem with randomness in Sony Playstation 3 (ECDSA signatures, 2010)

Later . . . in openssl 1.0 source code

```
/* 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. */
```

- Correct and secure implementation of cryptography is not easy
  - 10 vulnerabilities in openssl (NVD, published in 2010–2014) with severity High, not including the Heartbleed bug (with severity Medium)

Heartbleed

- probably the most important vulnerability in 2014
- problem in implementation of heartbeat extension (RFC6520) in OpenSSL
- the attacker can read the memory of the server

  request (as intended): send me this 6 byte payload “abcdef”
  response: “abcdef”

  request (attack): send me this 20003 byte payload “abc”
  response: “abc” + 20000 bytes of server’s memory

Timing attacks on comparisons (Google, Sun, …)

- 2009 – Keyczar (Google), Java (Sun), …
- common scenario: server compares received HMAC with calculated one
- attacker's goal: to get correct HMAC for his own message ("authentic")
- What is wrong with this code (Python)?

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

What is wrong with this code (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;
}
```

HMAC reconstruction

How long does it take for server to answer/react to incorrect HMAC

- if the first 0, 1, 8 or 15 bytes are correct?
- HMAC reconstruction based on time-variance of responses
- 4th byte:

```
71 A0 89 00 00 . . . 00
71 A0 89 01 00 . . . 00
         . . .
71 A0 89 4A 00 . . . 00      longer time to process?
         . . .
71 A0 89 FF 00 . . . 00
```

- usually multiple measures required for a single value (noise)
- statistical evaluation of measurements

Constant-time comparison (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;
}
```

Adobe password encryption

- 2013, Adobe
- data breach, 38 million *active* users account information exposed
- 150 million user accounts overall
- passwords are encrypted (the key was not leaked)
  …using 3DES (block cipher with 8 B block) in ECB mode
- result:
  - equal password substring [1-8], [9-16] easily identifiable
  - guess using password hits (part of account information), e.g.
    “numbers 123456”, “c’est 123456”
    “1*6”, “sixones”
    “q w e r t y”, “6 long qwert”

The most frequent passwords

source: Splashdata, based on leaked passwords (2015 and comparison with 2014)

1.	123456	
2.	password	
3.	12345678	(+ 1)
4.	qwerty	(+ 1)
5.	12345	(− 2)
6.	123456789	
7.	football	(+ 3)
8.	1234	(− 1)
9.	1234567	(+ 2)
10.	baseball	(− 2)
11.	welcome	(new)
12.	1234567890	(new)
13.	abc123	(+ 1)
14.	111111	(+ 1)
15.	1qaz2wsx	(new)
16.	dragon	(− 7)
17.	master	(+ 2)
18.	monkey	(− 6)
19.	letmein	(− 6)
20.	login	(new)
21.	princess	(new)
22.	qwertyuiop	(new)
23.	solo	(new)
24.	passw0rd	(new)
25.	starwars	(new)

WPS (WiFi Protected Setup)

- 2011
- goal: easy (and secure) method to add a device to network
- implementation:
  - 8 digit PIN code authentication (printed on a sticker)
  - theoretically 10^8 possibilities
  - practically: response to incorrect PIN leaks an information whether the first half of the PIN is wrong
    last digit is a checksum
  - $10^4 + 10^3$ possibilities
- WPS can't be turned off in some WiFi routers

Encrypted USB drives

- 2010; Kingstone, SanDisk, Verbatim
- FIPS 140-2 Level 2 certification; AES-256 encryption
- reality:
  - encryption key does not depend on user's password
  - USB key unlocks if some expected string (fixed, password- and device-independent) is received

Hash tables collisions

- 2011; Oracle, Microsoft, PHP, Apache Tomcat, ...
- analogous problem found originally in 2003; Perl, Squid
- hash table – data structure for storing (key/data) pairs
  - average complexity $O(n)$ for inserting/deleting/finding n elements
  - worst case complexity $O(n^2)$ for n elements (when keys collide)
- problem: colliding keys can be generated easily
- parameters of HTTP POST requests are parsed into hash table automatically
- DoS attack on web server:
  ~70-100kbits/s $\Rightarrow$ one i7 core busy (2011, PHP)

Hashing for hash tables

- Java 6 (java.lang.String, method public int hashCode())
  - 32-bit arithmetic (int), s_i denotes an i-th character of an $(s_{1,\ldots,n})$:

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

- PHP 5 (algorithm DJBX33A, 32-bit arithmetic), s_0 is constant 5381

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

- ASP.NET (algorithm DJBX33X), s_0 is constant 5381

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

- easy to find large multicollisions

Solutions

- limit the size of POST requests, limit CPU for single request, etc.
- better hash function
  - for example randomized hashing – the function dependent on randomly chosen parameter (when process starts)

Apple “goto” fail (2014)

```
SSLVerifySignedServerKeyExchange(...)
{
  OSStatus  err;
  ...
  if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
    goto fail;
  if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
    goto fail;
    goto fail;
  if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
    goto fail;
    ...
    err = sslRawVerify(...)

  fail:
    ...
    return err;
}
```

Other incidents (1)

- NSA
  - 2013; approx. 1.7 million files
  - Showden (contractor)
  - gradual publication of documents and files, global surveillance programs
    - tools and methods, e.g. see Tailored Access Operations (TAO) catalog
    - identities of cooperating companies and governments
    - identities of ISPs and platforms that NSA has penetrated or attempted to penetrate
    - foreign officials and systems that NSA has targeted
- Associated Press
  - April 2013
  - AP Twitter account hacked:
    Breaking: Two Explosions in the White House and Barack Obama is Injured.
  - 136 billion USD from the S&P's 500 Index in two minutes

Other incidents (2)

- Network Time Protocol – DoS attacks
  - NTP amplification attack (amplification factor 19)
  - single 234-byte request . . . 10 packets response (total 4 460 bytes).
  - MONLIST command (IP addresses of the last 600 machines interacting with an NTP server)
  - February 2014 . . . reported DDoS attack with 400 Gbps traffic
- BMW
  - ConnectedDrive technology (since 2010)
  - more than 50 models affected (BMW, Mini, Rolls Royce)
  - features: emergency call, remote unlocking and unlocking, auxiliary heating, etc.
  - analysis in 2014, published in 2015
  - the same symmetric keys in all cars
  - some services do not encrypt messages between the car and BMW's servers
  - device is not tamper-proof, etc.

Other incidents (3)

- RSA
  - 2011; targeted attack on RSA (part of EMC)
  - SecurID tokens (one-time passwords, two-factor authentication), market leader
  - replacing 40 million tokens
- NIST, RSA
  - 2013
  - NIST standard includes Dual Elliptic Curve Deterministic Random Bit Generator (Dual EC DRBG)
  - Dual EC DRBG proposed by NSA
  - RSA: Dual EC DRBR default in BSAFE toolkit
  - RSA: 10 million USD deal with NSA (Reuters)
  - trapdoor in Dual EC DRBR