

# Lập trình an toàn trong Java



cuu duong than cong . com

cuu duong than cong . com

# Tổng quan

cuu duong than cong . com

cuu duong than cong . com

## Nội dung

---

- Java và tầm quan trọng
- Java security và thách thức
- CVE
- Nguyên tắc

## Lỗi phần mềm nghiêm trọng

28/6/1962: Tàu thăm dò vũ trụ Mariner I bị phá hủy

1988: Lỗi tràn bộ đệm trong trình Berkeley Unix tạo ra worm

4/6/1996: Ariane 5 Flight 501 phát nổ chỉ 40 giây sau khi phóng



Bảo mật web và ứng dụng - ATTT - UIT

4

- 28/6/1962, Tàu thăm dò vũ trụ Mariner I:  
Có sứ mệnh bay đến sao Kim nhưng đã bị phá hủy 293 giây sau khi phóng do bay chệch hướng so với dự kiến ban đầu.  
Nguyên nhân: một công thức được viết trên giấy bằng bút chì đã không được chuyển đổi, khiến hệ thống máy tính tính toán sai đường đi của tên lửa → **cấu tả trong lập trình**
- 1988, Lỗi tràn bộ đệm trong trình Berkeley Unix:  
Morris, sâu Internet đầu tiên, đã lây lan tới khoảng 2.000 - 6.000 máy tính của nhiều tổ chức ở Mỹ trong vòng chưa đầy một ngày bằng cách tận dụng **lỗi tràn bộ nhớ** trung gian. Hàm `get()` được thiết kế để nhận luồng text qua mạng nhưng nó không được dự tính trước để hạn chế khối lượng nhập. Tận dụng khiếm khuyết này, một lượng lớn dữ liệu đã được gửi tới, tạo điều kiện cho phép sâu kiểm soát bất cứ hệ thống nào nó có thể kết nối. Các chuyên gia đã quyết định loại `get()` trong mã. Tuy nhiên, họ từ chối xóa `get()` trong thư viện nhập/xuất chuẩn của ngôn ngữ lập trình C và nó vẫn còn tồn tại đến ngày nay.
- 4/6/1996, Ariane 5 Flight 501:  
Mã hoạt động cho tên lửa Ariane 4 được sử dụng lại trong phiên bản tiếp theo

Ariane 5. Tuy nhiên, rắc rối đã xảy ra khi mã này thực hiện quá trình **chuyển đổi số chứa dấu phẩy động 64 bit sang ký hiệu số nguyên 16 bit**. Động cơ trong Ariane 5 có tốc độ nhanh hơn đã khiến các số 64 bit trở lên lớn hơn so với Ariane 4, gây tình trạng quá tải và sập máy tính điều khiển. Hệ thống backup cũng gặp trục trặc ngay 0,05 giây sau đó và thiết bị xử lý cơ sở của tên lửa bị tiếp quá nhiều năng lượng. Ariane 5 đã phát nổ chỉ 40 giây sau khi phóng.

cuu duong than cong . com

cuu duong than cong . com

## Tầm quan trọng của bảo mật ứng dụng

- Desktop Application có được sử dụng nhiều như ứng dụng web?
- Desktop Application có thật sự quan trọng để tấn công khi mà internet phát triển mạnh?

Bảo mật web và ứng dụng - ATTT - UIT

5

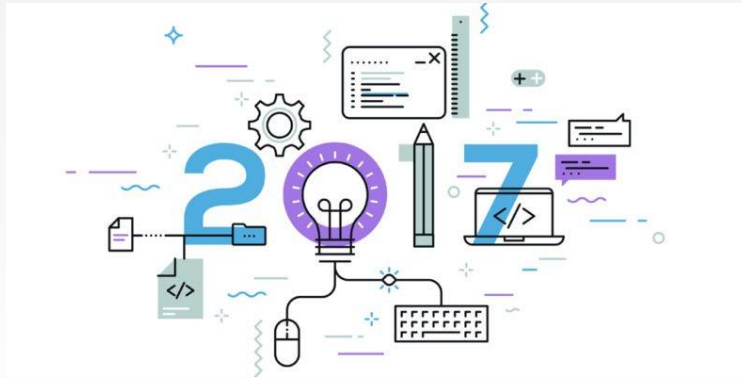
- Ứng dụng Desktop:

Sử dụng nhiều trong các cơ quan tổ chức như ứng dụng thống kê, kế toán, quản lý thông tin xuất nhập khẩu, bệnh viện, ngân hàng, cơ quan nhà nước,... Thường sử dụng nội bộ → đảm bảo tính bảo mật, thao tác nhanh.

Hạn chế việc nhân viên truy cập vào internet → dễ bị tấn công, tránh sao nhãng công việc.

- Thông tin nhạy cảm cao → mất dữ liệu, mất tiền, bí mật quốc gia, quân sự, quốc phòng, khoa học kỹ thuật,...

## Top 10 ngôn ngữ lập trình phổ biến



Bảo mật web và ứng dụng - ATTT - UIT

6

cuu duong than cong . com

cuu duong than cong . com

## Top 10 ngôn ngữ lập trình phổ biến

(Tính đến tháng 6 năm 2017)



No. 1



No. 2



No. 3



No. 4



No. 5



No. 6



No. 7



No. 8



No. 9



No. 10

Nguồn: IEEE

<https://spectrum.ieee.org/computing/software/the-2017-top-programming-languages>

Bảo mật web và ứng dụng - ATTT - UIT

7

1. Python: dễ đọc, ít dòng lệnh hơn C, Java; thư viện lớn.
2. C: hỗ trợ lập trình hệ thống, đa nền tảng, rõ ràng, khả năng truy cập phần cứng và có thể tạo ra mã nhị phân kích thước nhỏ.
3. Java: Hướng đối tượng và theo nguyên tắc WORA: write once, run anywhere. Lựa chọn hàng đầu của các dev.
4. C++: thao tác bộ nhớ ở mức thấp, có thể bắt nhiều lỗi trong thời gian biên dịch, hiệu suất tốt.
5. C#: OOP.
6. R: nguồn mở, ngôn ngữ cho thống kê.
7. Javascript: Xử lý web client-side, có xu hướng phát triển mạnh.
8. PHP server-side, tính khả chuyển, có nhiều framework và giải pháp
9. Go:
  - Khai thác sức mạnh của các bộ xử lý đa lõi và phần cứng thể hệ mới; đối với các ngôn ngữ hiện có C, Java,... → không thể.
  - Ngôn ngữ mới do Google thiết kế và phát triển, kỳ vọng sẽ giúp ngành công nghiệp phần mềm khai thác nền tảng đa lõi của bộ vi xử lý và hoạt động đa nhiệm tốt hơn. (mạng lưới máy)
  - Thay vì chỉ HĐH được cấp tài nguyên và xử lý thì giờ phần mềm cũng có thể tương tác trực tiếp với nền tảng đa lõi.



10. Swift: IOS, đang phát triển khá mạnh.

cuu duong than cong . com

cuu duong than cong . com

# Top 10 ngôn ngữ lập trình phổ biến

Top 10 Programming Languages										
	Python	C	Java	C++	C#	R	JavaScript	PHP	Go	Swift
Paradigm	Multi-paradigm: object-oriented, imperative, functional, procedural, reflective	Imperative (procedural), structured	Multi-paradigm: object-oriented, class-based, structured, imperative, generic, reflective, concurrent	Multi-paradigm: procedural, functional, object-oriented, generic	Multi-paradigm: structured, imperative, class-oriented, event-driven, task-driven, functional, generic, reflective, concurrent	Multi-paradigm: array, object-oriented, imperative, functional, procedural, reflective	Multi-paradigm: object-oriented, prototype-based, imperative, functional, event-driven	Imperative, object-oriented, procedural, reflective	Compiled, concurrent, imperative, structured	Multi-paradigm: protocol-oriented, object-oriented, functional, imperative, block-structured
Designed by	Guido van Rossum	Dennis Ritchie	James Gosling	Bjarne Stroustrup	Microsoft	Ross Ihaka and Robert Gentleman	Brendan Eich	Rasmus Lerdorf	Robert Griesemer, Rob Pike, Ken Thompson	Chris Lattner and Apple Inc.
Developer	Python Software Foundation	Dennis Ritchie & Bell Labs (AT&T, UNIX, BSD, I/O/EC)	Sun Microsystems (now owned by Oracle corporation)	Bell Labs	Microsoft	R Core Team	Netscape Communications Corporation, Mozilla Foundation, Iarna International	The PHP Development Team, Zend Technologies	Google Inc.	Apple Inc.
First appeared	20 February 1991 (28 years ago)	1972 (45 years ago)	May 23 1995 (22 years ago)	1983 (34 years ago)	2000 (17 years ago)	August 1993 (24 years ago)	December 4, 1995 (21 years ago)	June 8, 1995 (22 years ago)	November 10, 2009 (7 years ago)	June 2, 2014 (3 years ago)
Typing discipline	Duck, dynamic, strong	Static, weak, manual, non-strict	Static, strong, safe, nominative, manual	Static, nominative, partially inferred	Static, dynamic, strong, safe, nominative, partially inferred	Dynamic	Dynamic, duck	Dynamic, weak, gradual, as for PHP 7.0.0	Strong, static, inferred, structural	Static, strong, inferred
Platform	Cross platform	Cross platform	Windows, Solaris, Linux, OS X	Unix, MacOS, Solaris	Common Language Infrastructure	UNIX platforms, Windows, MacOS	Cross platform	Unix-like, Windows	Unix, macOS, FreeBSD, NetBSD, OpenBSD, Windows, Plan 9, Raspberry Pi OS, SUSE	Darwin, Linux, FreeBSD
Filename extensions	.py, .pyc, .pyo (later to 3.5: .pyw and .pyz)	.c, .h	.java, .class, .jar	.cc, .cpp, .c, .c++, .h, .hpp, .hxx, .h++	.cs	.r, .R, .RData, .rds, .rda	.js	.php, .php1, .php3, .php4, .php5, .php7, .phpn	.go	.swift
















Bảo mật web và ứng dụng - ATTT - UIT

8

cuu duong than cong . com

cuu duong than cong . com

## Top 10 ngôn ngữ lập trình phổ biến

Language Rank	Types	Spectrum Ranking
1. Python	 	100.0
2. C	  	99.7
3. Java	  	99.5
4. C++	  	97.1
5. C#	  	87.7
6. R		87.7
7. JavaScript	 	85.6
8. PHP		81.2
9. Go	 	75.1
10. Swift	 	73.7

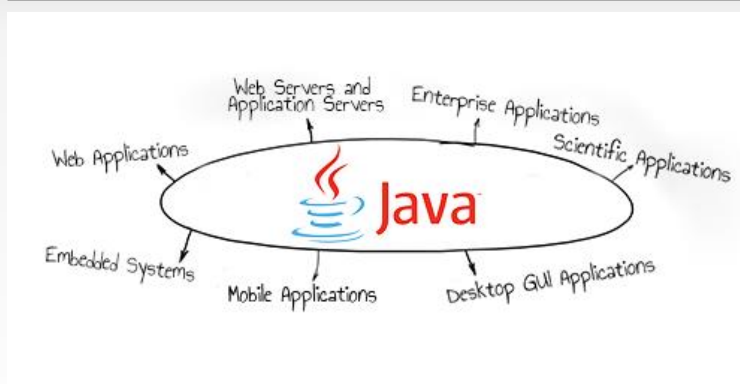
Bảo mật web và ứng dụng - ATTT - UIT

9

cuu duong than cong . com

cuu duong than cong . com

# Ngôn ngữ lập trình Java



Nguồn:

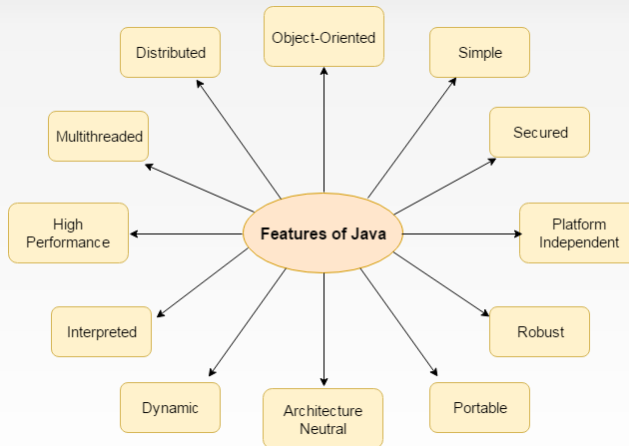
<https://www.invensis.net/blog/it/applications-java-programming-language/>

Bảo mật web và ứng dụng - ATTT - UIT

10

- Phát triển bởi Sun Microsystems 1995, Oracle mua lại 2010.
  - Xương sống của hàng triệu ứng dụng thuộc nhiều nền tảng khác nhau: Windows, Macintosh, OS dựa trên nhân UNIX, di động trên nền Android, hệ thống nhúng, giải pháp thương mại.
  - Hiện chạy trên hơn 3 triệu thiết bị.
1. Desktop GUI App:
    - AWT - Abstract Widowing Toolkit: menu, btn, list,...
    - Swing: tree, table, scroll pane, list
    - JavaFX: graphic, media
  2. Java ME, J2ME: phone, smart phone, Android (SDK).
  3. Embedded: 100% thiết bị chơi nhạc sử dụng đĩa Blu-ray và 125 triệu TV dùng Java - Oracle
  4. Web App:
    - Servlets, Struts or JSPs: giáo dục, chính phủ, y tế, tài chính
    - eCommerce: Broadleaf
  5. Web server: Tomcat
  6. Enterprise App: Java Enterprise Edition (Java EE) cung cấp API, runtime environment cho ứng dụng mạng và dịch vụ web.
  7. Scientific App: Matlab

# Ngôn ngữ lập trình Java



Bảo mật web và ứng dụng - ATTT - UIT

11

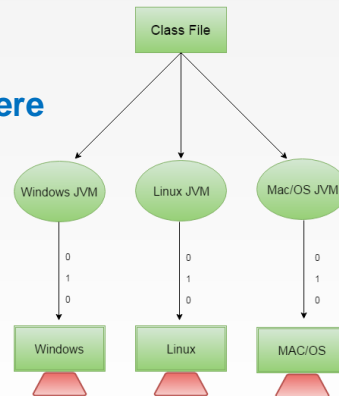
1. Đơn giản: cú pháp dựa trên C++, bỏ đi những đặc trưng ít dùng, gây rối như:
  - Khai báo con trỏ
  - Quá tải toán tử,
  - Không cần xóa đối tượng không có tham chiếu nhờ có cơ chế dọn rác.
2. OOP: giúp phát triển và duy trì dễ dàng.

## Độc lập nền tảng

Nền tảng (platform) là gì?

Có mấy loại?

**Write Once, Run Anywhere**



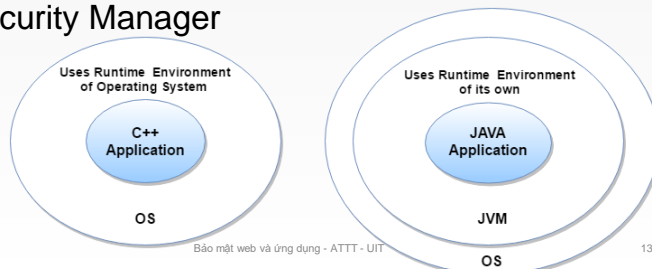
Bảo mật web và ứng dụng - ATTT - UIT

12

- **Nền tảng** là môi trường phần cứng hoặc phần mềm cho chương trình chạy.
  - Java cung cấp nền tảng dựa trên phần mềm, có thể chạy trên những nền tảng phần cứng khác nhau, gồm 2 thành phần: runtime environment và API.
- Mã Java được biên dịch thành bytecode, độc lập nền tảng.  
Win, Linux, Mac, Sun, Solaris.

## Bảo mật

- No explicit pointer
- Chương trình Java chạy trong VM sandbox
- Classloader: tách riêng package
- Bytecode Verifier
- Security Manager



Bảo mật web và ứng dụng - ATTT - UIT

13

**Classloader:** Tách biệt package cho những class của hệ thống (file local) với những class được import từ tài nguyên mạng.

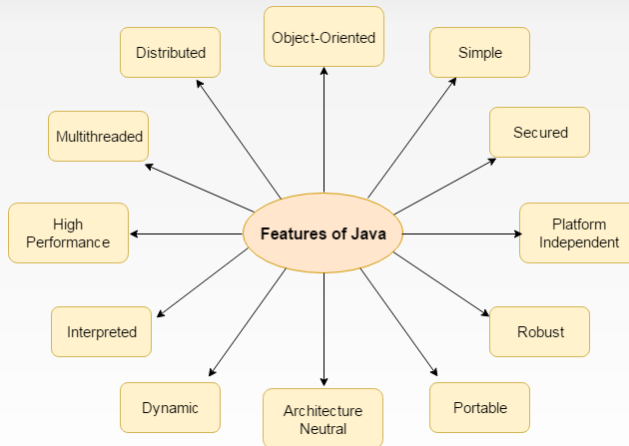
**Bytecode Verifier:** kiểm tra code bất hợp pháp, vi phạm quyền truy cập đến obj.

**Security Manager:** xác định tài nguyên 1 class có thể truy cập như đọc, ghi trên đĩa local.

Được cung cấp bởi Java.

Cung cấp bởi developer: SSL, JAAS (*Java Authentication and Authorization Service*), Crypto.

# Ngôn ngữ lập trình Java



Bảo mật web và ứng dụng - ATTT - UIT

14

1. Đơn giản: cú pháp dựa trên C++, bỏ đi những đặc trưng ít dùng, gây rối như:
  - Khai báo con trỏ
  - Quá tải toán tử,
  - Không cần xóa đối tượng không có tham chiếu nhờ có cơ chế dọn rác.
2. OOP: giúp phát triển và duy trì dễ dàng.
3. Robust: mạnh mẽ
  - Cơ chế quản lý bộ nhớ.
  - Không có con trỏ nên tránh vấn đề bảo mật.
  - Có cơ chế thu gom rác tự động.
  - Xử lý ngoại lệ và cơ chế kiểm tra type.
4. Kiến trúc trung lập: không có tính năng bị phụ thuộc, như kích thước của type nguyên thủy được cố định. Ngôn ngữ C dữ liệu int chiếm 2 byte trên kiến trúc 32-bit, 4 byte trên 64bit. Java luôn là 4byte.
5. Khả chuyển: viết 1 lần chạy trên nhiều nền tảng desktop, mobile, hệ thống nhúng (bytecode).
6. Hiệu suất cao: nhanh hơn biên dịch truyền thống vì bytecode gần mã tự nhiên nhưng vẫn chậm hơn ngôn ngữ được biên dịch như C++.
7. Phân tán: tạo ứng dụng phân tán. RMI (*Remote Method Invocation*) và EJB (*Enterprise JavaBeans*) được dùng để tạo ứng dụng phân tán. Có thể truy cập tập



tin bằng cách gọi phương thức từ bất kỳ máy nào trên internet.

8. Đa luồng: 1 luồng như 1 chương trình, thực thi đồng thời. Ưu điểm chính là không chiếm bộ nhớ cho mỗi luồng, mà chia sẻ khu vực bộ nhớ.

Ngoài ra:

- Miễn phí
- Ngôn ngữ phổ biến, có lượng cộng đồng người dùng năng động và hỗ trợ.
- Công cụ phát triển mạnh mẽ: Eclipse SDK và NetBean, khả năng debug và môi trường phát triển tích hợp.
- Tăng tính đa dạng của ngôn ngữ, tương thích với Scala, Groovy, Jruby, clojure.
- Khả năng tương thích phiên bản (bản sau có thể chạy được bản trước).
- Java 8 đem đến nhiều tính năng như nền tảng với khả năng mở rộng và linh hoạt cho IOT, API, thư viện DateTime mới, công cụ đồ họa, tích hợp JS,...

cuu duong than cong . com

cuu duong than cong . com

## Lập trình an toàn

---

- Bảo mật là một phần trong quá trình phát triển phần mềm → đảm bảo tính bảo mật, toàn vẹn và khả dụng.
- Phần mềm an toàn là kết quả của quá trình phát triển phần mềm nhận thức về sự bảo mật
  - Bảo mật được tích hợp
  - Được phát triển với ý thức bảo mật.

Bảo mật web và ứng dụng - ATTT - UIT

15

Quá trình liên tục, liên quan đến con người và thực tiễn.

## Lập trình an toàn (tt)

---

- Bảo mật hiệu quả nhất nếu được lập kế hoạch và quản lý:
  - Trong suốt giai đoạn phát triển phần mềm (SDLC)
  - Trong ứng dụng quan trọng hoặc quá trình xử lý các thông tin nhạy cảm.
- **Giải pháp bảo mật** trong phát triển phần mềm **không chỉ là công nghệ**.

Bảo mật web và ứng dụng - ATTT - UIT

16

Software development life cycle.

## Thách thức

- Ứng dụng, hệ thống, mạng chịu sự tấn công liên tục: mã độc, từ chối dịch vụ
  - Virus
  - Trojan horse
  - Worm
  - Applet
  - ...
- Ứng dụng có thể chứa các lỗ hổng bảo mật
- Tăng bảo mật → tăng chi phí

Bảo mật web và ứng dụng - ATTT - UIT

17

Lỗ hổng do vô tình hay cố ý của developer.

Sự kiểm soát phần mềm, môi trường, phần cứng được yêu cầu dù không thể ngăn cản những vấn đề được tạo ra bởi thực tiễn lập trình yếu kém.

## Kiểm tra bảo mật

---

- Đảm bảo hệ thống ngăn chặn người dùng trái phép truy cập tài nguyên và dữ liệu
- Các thuộc tính thông thường:
  - Tính xác thực
  - Tính toàn vẹn
  - Quyền
  - Không từ chối
  - Tính bảo mật
  - Khả năng phục hồi
  - Tính khả dụng

# CVE

## Common Vulnerabilities and Exposures

- ✓ Cung cấp thông tin về các lỗ hổng bảo mật
- ✓ Giúp đánh giá sơ bộ công cụ bảo mật

Thống kê:

- Theo năm
- Theo loại
- Sản phẩm

...

Bảo mật web và ứng dụng - ATTT - UIT

19

CVE là:

- + Một định danh cho 1 lỗ hổng
- + Một mô tả được chuẩn hóa cho mỗi lỗ hổng
- + Một từ điển hơn 1 csdl
- + Cách CSDL và công cụ khác loại có thể hoạt động cùng ngôn ngữ
- + Nền tảng để đánh giá giữa công cụ và csdl
- + Miễn phí download và sử dụng

Ngoài ra còn có: OWASP, CWE, CERT, SAMATE

# 

### 

[Vulnerabilities \(499\)](#)
[CVSS Scores Report](#)
[Browse all versions](#)
[Possible matches for this product](#)
[Related Metasploit Modules](#)

[Related OVAL Definitions](#) : 
 [Vulnerabilities \(839\)](#)
[Patches \(556\)](#)
[Inventory Definitions \(3\)](#)
[Compliance Definitions \(0\)](#)

[Vulnerability Feeds & Widgets](#)

#### 

Year	# of Vulnerabilities	DoS	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass something	Gain Information	Gain Privileges	CSRF	File Inclusion	# of exploits
2010	1		1												
2011	3														
2012	59	3	1							2					
2013	180	1	10	4	4		1			32					2
2014	115	1	1												
2015	80														
2016	37		1	1							1	1			
2017	24	1									1				
Total	499	6	14	5	4		1			34	2	1			2
% Of All		1.2	2.8	1.0	0.8	0.0	0.2	0.0	0.0	6.8	0.4	0.2	0.0	0.0	

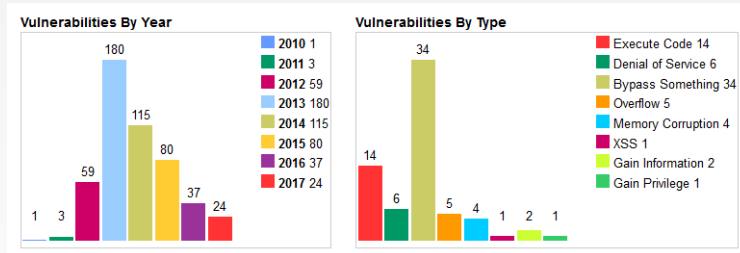
Bảo mật web và ứng dụng - ATTT - UIT

20

cuu duong than cong . com

cuu duong than cong . com

## Thống kê lỗ hổng JRE - Oracle



Bảo mật web và ứng dụng - ATTT - UIT

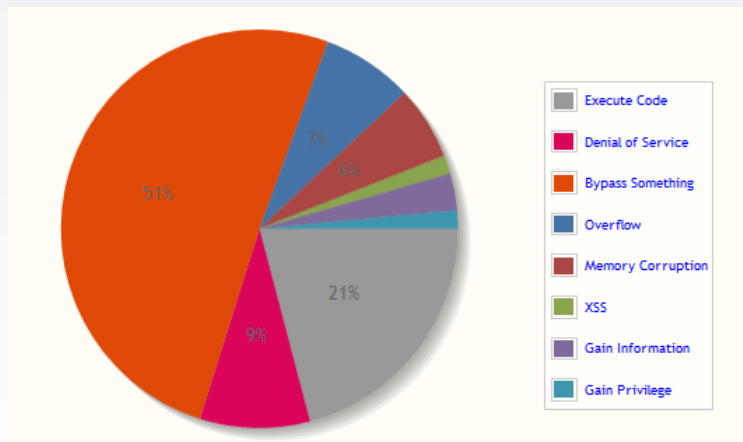
21

cuu duong than cong . com

cuu duong than cong . com



## Thống kê lỗi hỏng JRE - Oracle



Bảo mật web và ứng dụng - ATTT - UIT

22

cuu duong than cong . com

cuu duong than cong . com

# Thống kê lỗ hổng JRE - Oracle

## Oracle » JRE : Security Vulnerabilities

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

Total number of vulnerabilities : 499 Page : 1 (This Page) 2 3 4 5 6 7 8 9 10

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-2017-3546	284			2017-04-24	2017-07-12	4.3	None	Remote	Medium	Not required	None	Partial	None
Vulnerability in the Java SE, Java SE Embedded, JRockit component of Oracle Java SE (subcomponent: Networking). Supported versions that are affected are Java SE: 6u141, 7u131 and Bu121; Java SE Embedded: Bu121; JRockit: R28.3.13. Difficult to exploit vulnerability allows unauthenticated attacker with network access via SMTP to compromise Java SE, Java SE Embedded, JRockit. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Java SE, Java SE Embedded, JRockit accessible data. Note: Applies to client and server deployment of Java. This vulnerability can be exploited through sandboxed Java Web Start applications and sandboxed Java applets. It can also be exploited by supplying data to APIs in the specified Component without using sandboxed Java Web Start applications or sandboxed Java applets, such as through a web service. CVSS 3.0 Base Score 3.7 (Integrity impacts). CVSS Vector: (CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N).														
2	CVE-2017-3539	284			2017-04-24	2017-07-10	5.1	None	Remote	High	Single system	None	Partial	None
Vulnerability in the Java SE, Java SE Embedded component of Oracle Java SE (subcomponent: Security). Supported versions that are affected are Java SE: 6u141, 7u131 and Bu121; Java SE Embedded: Bu121; JRockit: R28.3.13. Difficult to exploit vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Java SE, Java SE Embedded, JRockit. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Java SE, Java SE Embedded, JRockit accessible data. Note: Applies to client and server deployment of Java. This vulnerability applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the Internet) and rely on the Java sandbox for security. This vulnerability does not apply to Java deployments, typically in servers, that load and run only trusted code (e.g., code installed by an administrator). CVSS 3.0 Base Score 3.1 (Integrity impacts). CVSS Vector: (CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N).														
3	CVE-2017-3533	284			2017-04-24	2017-07-10	4.3	None	Remote	Medium	Not required	None	Partial	None
Vulnerability in the Java SE, Java SE Embedded, JRockit component of Oracle Java SE (subcomponent: Networking). Supported versions that are affected are Java SE: 6u141, 7u131 and Bu121; Java SE Embedded: Bu121; JRockit: R28.3.13. Difficult to exploit vulnerability allows unauthenticated attacker with network access via FTP to compromise Java SE, Java SE Embedded, JRockit. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Java SE, Java SE Embedded, JRockit accessible data. Note: Applies to client and server deployment of Java. This vulnerability can be exploited through sandboxed Java Web Start applications and sandboxed Java applets. It can also be exploited by supplying data to APIs in the specified Component without using sandboxed Java Web Start applications or sandboxed Java applets, such as through a web service. CVSS 3.0 Base Score 3.7 (Integrity impacts). CVSS Vector: (CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N).														
4	CVE-2017-3526	284			2017-04-24	2017-07-10	7.4	None	Remote	Medium	Not required	None	None	Complete
Vulnerability in the Java SE, Java SE Embedded, JRockit component of Oracle Java SE (subcomponent: JAXP). Supported versions that are affected are Java SE: 6u141, 7u131 and Bu121; Java SE Embedded: Bu121; JRockit: R28.3.13. Difficult to exploit vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Java SE, Java SE Embedded, JRockit. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently crash complete DOS of Java SE, Java SE Embedded, JRockit. Note: Applies to client and server deployment of Java. This vulnerability can be exploited through sandboxed Java Web Start applications and sandboxed Java applets. It can also be exploited by supplying data to APIs in the specified Component without using sandboxed Java Web Start applications or sandboxed Java applets, such as through a web service. CVSS 3.0 Base Score 5.9 (Availability impacts). CVSS Vector: (CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:N/A:H).														

# Thống kê lỗ hổng JRE - Oracle

## Vulnerability Details : [CVE-2017-3526](#)

Vulnerability in the Java SE, Java SE Embedded, JRockit component of Oracle Java SE (subcomponent: JAXP). Supported versions that are affected are Java SE: 6u141, 7u131 and 8u121; Java SE Embedded: 8u121; JRockit: R28.3.13. Difficult to exploit vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Java SE, Java SE Embedded, JRockit. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of Java SE, Java SE Embedded, JRockit. Note: Applies to client and server deployment of Java. This vulnerability can be exploited through sandboxed Java Web Start applications and sandboxed Java applets. It can also be exploited by supplying data to APIs in the specified Component without using sandboxed Java Web Start applications or sandboxed Java applets, such as through a web service. CVSS 3.0 Base Score 5.9 (Availability impacts). CVSS Vector: (CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:N/A-H).  
Publish Date : 2017-04-24 Last Update Date : 2017-07-10

[Collapse All](#) [Expand All](#) [Select](#) [Select/Copy](#) [Scroll To](#) [Comments](#) [External Links](#)  
[Search Twitter](#) [Search YouTube](#) [Search Google](#)

### CVSS Scores & Vulnerability Types

CVSS Score **5.9**  
Confidentiality Impact **None** (There is no impact to the confidentiality of the system.)  
Integrity Impact **None** (There is no impact to the integrity of the system.)  
Availability Impact **Complete** (There is a total shutdown of the affected resource. The attacker can render the resource completely unavailable.)  
Access Complexity **Medium** (The access conditions are somewhat specialized. Some preconditions must be satisfied to exploit.)  
Authentication **Not required** (Authentication is not required to exploit the vulnerability.)  
Gained Access **None**  
Vulnerability Type(s) **288**  
CWE ID **288**

### Related OVAL Definitions

#### Products Affected By CVE-2017-3526

#	Product Type	Vendor	Product	Version	Update	Edition	Language
1	Application	Oracle	JDK	1.6	Update 141		<a href="#">Version Details Vulnerabilities</a>
2	Application	Oracle	JDK	1.7	Update 131		<a href="#">Version Details Vulnerabilities</a>
3	Application	Oracle	JDK	1.8	Update 121		<a href="#">Version Details Vulnerabilities</a>
4	Application	Oracle	JSE	1.6	Update 141		<a href="#">Version Details Vulnerabilities</a>
5	Application	Oracle	JSE	1.7	Update 131		<a href="#">Version Details Vulnerabilities</a>
6	Application	Oracle	JSE	1.8	Update 121		<a href="#">Version Details Vulnerabilities</a>
7	Application	Oracle	JRockit	R28.3.13			<a href="#">Version Details Vulnerabilities</a>

Bảo mật web và ứng dụng - ATTT - UIT

24

cuu duong than cong . com

cuu duong than cong . com

# Nguyên tắc

cuu duong than cong . com

cuu duong than cong . com

## Giới thiệu

Java cung cấp một môi trường an toàn cho việc thực thi mã nguồn

- Kiến trúc bảo mật Java:
  - Bảo vệ người dùng và hệ thống từ những chương trình được tải qua mạng
  - Không thể bảo vệ khỏi bug trong thực thi, tạo lỗ hổng truy cập:
    - ✓ Tập tin
    - ✓ Máy in
    - ✓ Webcam
    - ✓ Micro
    - ✓ Mạng

An toàn thông tin - UIT

26

- Một trong những cân nhắc thiết kế chính cho nền tảng Java là cung cấp một môi trường an toàn để thực thi mã nguồn.
- Kiến trúc bảo mật Java có thể bảo vệ người dùng và hệ thống từ các chương trình có hại tải qua mạng, nó không thể bảo vệ chống lại lỗi thực thi xuất hiện trong mã đáng tin cậy.
- Một vài trường hợp chương trình cục bộ có thể được thực thi, bảo mật Java bị tắt.  
→ Bug có thể biến máy tính thành máy tính zombie, đánh cắp dữ liệu từ máy tính + mạng cục bộ, gián điệp thông qua các thiết bị kèm theo, ngăn chặn hoạt động hữu ích của máy, hỗ trợ tấn công và nhiều hoạt động độc hại khác.

## Giới thiệu (tt)

- Ngôn ngữ và máy ảo Java cung cấp nhiều tính năng → giảm thiểu lỗi lập trình phổ biến.
  - Loại ngôn ngữ an toàn và runtime cung cấp cơ chế quản lý bộ nhớ tự động và kiểm tra giới hạn.
  - Chương trình và thư viện kiểm tra trạng thái không hợp lệ tại các thời điểm sớm nhất

An toàn thông tin - UIT

27

Việc lựa chọn ngôn ngữ lập trình tác động đến khả năng của chương trình.  
→ chống tấn công tràn bộ đệm (C, C++).

## Nguyên tắc cơ bản

---

1. Có những thiếu sót rõ ràng hơn có những thiếu sót không rõ ràng
2. Thiết kế các API để tránh những vấn đề liên quan đến bảo mật
3. Tránh sao chép
4. Hạn chế những đặc quyền

An toàn thông tin - UIT

28

- Java có thể giúp lấp những thiếu sót rõ ràng.
- Tốt hơn là thiết kế những API mới với ý định bảo mật hơn là cố bổ sung bảo mật vào một API đã có.
- Code và data được sao chép có xu hướng không được đối xử như code nguồn → gây ra rắc rối vì những thay đổi thường không được sao chép.
- Có review kĩ vẫn không thể loại bỏ hết thiếu sót. Nếu đặc quyền bị hạn chế → khai thác lỗ hổng bị hạn chế (Nguyên tắc quyền tối thiểu).

## Nguyên tắc cơ bản (tt)

5. Thiết lập những ranh giới tin cậy
6. Giảm thiểu số lần kiểm tra quyền
7. Đóng gói
8. Cung cấp tài liệu thông tin liên quan đến bảo mật

An toàn thông tin - UIT

29

- Data truyền qua những ranh giới tin cậy nên được sanitize và validate trước khi dùng, những mã nguồn đảm bảo tính toàn vẹn của ranh giới tin cậy phải được load trong cách được bảo đảm. Khi kiểm tra những ranh giới tin cậy, nên nhớ một vài câu hỏi sau:
  - + Code và data đã sử dụng đủ tin cậy?
  - + Một thư viện có thể bị thay thế với 1 cài đặt độc hại?
  - + Dữ liệu cấu hình không tin cậy có đang được sử dụng?
  - + Code đang gọi với đặc quyền thấp hơn có đủ bảo vệ phòng chống không?
- Java là ngôn ngữ đối tượng-khả năng (object-capability). Kiểm tra *SecurityManager* nên là cách cuối cùng. Thực hiện kiểm tra bảo mật tại một vài điểm được xác định và trả về một đối tượng (một khả năng) mà client code giữ lại để không có kiểm tra quyền nào nữa được yêu cầu.
- Đóng gói: sắp xếp các cách xử lý (hành vi) và cung cấp giao diện đơn giản. Field của các đối tượng nên private (bí mật) và tránh truy cập từ bên ngoài. Giao diện của phương thức, lớp, package và mô-đun nên tạo thành một tập chặt chẽ của các hành vi.
- Tài liệu API nên chứa các thông tin liên quan bảo mật như quyền được yêu cầu, trường hợp ngoại lệ liên quan bảo mật, bộ gọi nhạy cảm, và bất kỳ điều kiện có liên quan đến bảo mật. Cung cấp thông tin trong chú thích cho tool



như javadoc cũng có thể giúp đảm bảo rằng nó được cập nhật.

cuu duong than cong . com

cuu duong than cong . com

## Tóm tắt

---

**Không nên dựa vào  
các tính năng bảo mật  
để loại bỏ các khiếm khuyết an ninh.**

An toàn thông tin - UIT

30

cuu duong than cong . com

cuu duong than cong . com